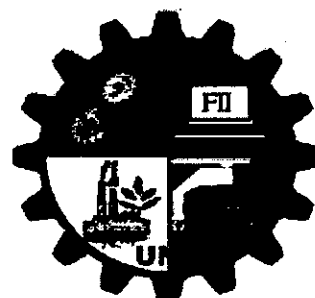


**UNIVERSIDAD NACIONAL DE PIURA**  
**FACULTAD DE INGENIERIA INDUSTRIAL**  
**ESCUELA PROFESIONAL DE INGENIERIA INFORMATICA**



**“CREACION DE UNA APLICACIÓN DE REALIDAD  
AUMENTADA PARA MEJORAR LAS VENTAS EN LA EMPRESA  
IMPORTACIONES LABAN SAC- HUANCABAMBA”**

**Presentada por:**

**Eysson Francisco Saucedo García**

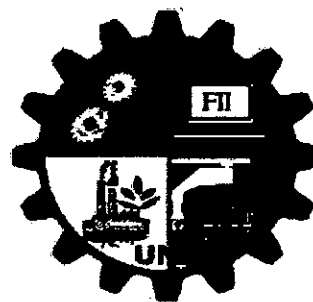
**TESIS PARA OPTAR EL TITULO DE  
INGENIERO INFORMATICO**

**Piura, Perú**

**2015**

777  
SAU

Tesis presentada como requisito para optar el título de Ingeniero Informático



Asesor:

A handwritten signature in black ink, appearing to read "V. Valle Ríos", written over a horizontal dashed line.

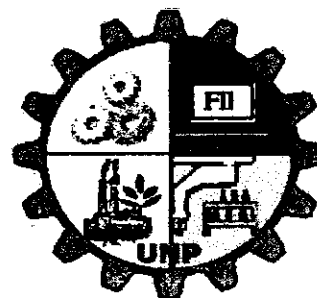
Ing. Víctor Hugo Valle Ríos

Tesista:

A handwritten signature in black ink, appearing to read "Eysson F. Saucedo", written over a horizontal dashed line.

Br. Eysson Francisco Saucedo García

**UNIVERSIDAD NACIONAL DE PIURA**  
**FACULTAD DE INGENIERIA INDUSTRIAL**  
**ESCUELA PROFESIONAL DE INGENIERIA INFORMATICA**



**“CREACION DE UNA APLICACIÓN DE REALIDAD AUMENTADA  
PARA MEJORAR LAS VENTAS EN LA EMPRESA  
IMPORTACIONES LABAN SAC- HUANCABAMBA”**

**Tesista: Eysson Francisco Saucedo García**

**Asesor: Ing. Víctor Hugo Valle Ríos, MSc**

**Miembros del jurado calificador**

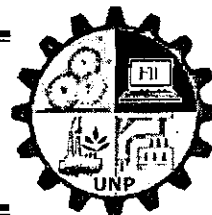
Ing. HECTOR WILMER FIESTAS BANCAYÁN, MSc  
PRESIDENTE - JURADO CALIFICADOR

Ing. PEDRO ANTONIO CRIOLLO GONZALES, MSc  
VOCAL - JURADO CALIFICADOR

Ing. ARTURO SANDOVAL RIVERA  
SECRETARIO - JURADO CALIFICADOR



UNIVERSIDAD NACIONAL DE PIURA  
FACULTAD DE INGENIERÍA INDUSTRIAL  
DECANATO



## ACTA DE SUSTENTACIÓN DE TESIS

Los Miembros del Jurado Calificador Ad-Hoc de la Tesis denominada: «**CREACIÓN DE UNA APLICACIÓN DE REALIDAD AUMENTADA PARA MEJORAR LAS VENTAS EN LA EMPRESA IMPORTACIONES LABAN S.A.C. - HUANCABAMBA**», presentada por el señor **EYSSON FRANCISCO SAUCEDO GARCÍA**, Bachiller de la Escuela Profesional en Ingeniería Informática; asesorada por el **Ing. Víctor Hugo Valle Ríos, MSc.**; reunidos para la sustentación de ésta y luego de escuchar su exposición y las respuestas a las preguntas formuladas, la declaran:



Con el Calificativo:

Aprobado

Buena

En consecuencia el sustentante se encuentra apto para recibir el título profesional de **INGENIERO INFORMÁTICO**, conforme a Ley.

Piura, 20 de junio del 2015

Ing. HÉCTOR WILMER FIESTAS BANCAYÁN, MSc.  
PRESIDENTE – JURADO CALIFICADOR

Ing. PEDRO ANTONIO CRIOLLO GONZALES, MSc.  
VOCAL – JURADO CALIFICADOR

Ing. ARTURO SANDOVAL RIVERA  
SECRETARIO – JURADO CALIFICADOR



## INDICE:

### CONTENIDO

INDICE:	4
RESUMEN	10
INTRODUCCIÓN:	12
Capítulo I MARCO REFERENCIAL	13
I.1 PRESENTACIÓN DEL PROBLEMA	13
I.1.1 Descripción de la realidad problemática	13
I.1.2 Formulación del Problema	15
I.2 OBJETIVOS	16
I.2.1 Objetivo General	16
I.2.2 Objetivos Específicos	16
I.3 MARCO INSTITUCIONAL	17
I.4 MARCO TEORICO	17
I.5 HIPOTESIS:	40
I.5.1 Hipótesis de investigación	40
I.5.2 Operacionalización de las Variables	40
Capítulo II ANÁLISIS DE LA EMPRESA	41
II.1 ÁREA DE VENTAS	43
II.2 ANÁLISIS DEL SISTEMA DE VENTAS	44
Capítulo III BASE DE PARÁMETROS	47
III.1 SISTEMAS DE REALIDAD AUMENTADA A UTILIZAR	47
III.2 ACTIVACIÓN DE REALIDAD AUMENTADA	49
III.3 ELEMENTOS UTILIZADOS	51
Capítulo IV PLANIFICACIÓN	52
IV.1 PLANIFICACIÓN DE LOS REQUISITOS	53
IV.2 LISTA DE OBJETIVOS	54
IV.3 REQUISITOS PRIORIZADOS	54
IV.4 EJECUCIÓN DE ITERACIÓN	56
IV.5 INSPECCIÓN Y ADAPTACIÓN	56
IV.6 DEMOSTRACIÓN DE REQUISITOS	57



IV.6.1 Retrospectiva.....	57
Capítulo V IMPLEMENTACIÓN Y EVALUACIÓN DE RESULTADOS .....	58
V.1 IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA .....	58
V.1.1 Buscar la tecnología adecuada para dispositivos móviles (tablets y smartphone's) .....	58
V.1.2 Elección de herramientas para el desarrollo de la aplicación (Software).....	60
V.1.3 Creación del FrontEnd.....	74
V.1.4 Creación de Marcadores .....	76
V.1.5 Creación y búsqueda de objetos 3D .....	77
V.1.6 Creación del Backend.....	79
V.1.7 Prueba con marcadores impresos .....	94
V.2 CONSTATACIÓN DE RESULTADOS .....	101
CONCLUSIONES .....	106
RECOMENDACIONES .....	107
BIBLIOGRAFÍA .....	108
TESIS EN LA WEB .....	108
ARTICULOS EN INTERNET .....	109
ANEXOS .....	112

**Índice de Figuras:**

<i>Figura 1.</i>	Sistemas Operativos en el Mercado .....	20
<i>Figura 2.</i>	Arquitectura de Realidad Aumentada 1 .....	26
<i>Figura 3.</i>	Ejemplo Realidad Aumentada-Tablet .....	27
<i>Figura 4.</i>	Hololens .....	28
<i>Figura 5.</i>	Elementos que Comprenden la Realidad Aumentada .....	31
<i>Figura 6.</i>	Cuadro de Análisis de herramientas de desarrollo 3D .....	38
<i>Figura 7.</i>	Área Ventas de Importaciones Laban S.A.C – Huancabamba.....	43
<i>Figura 8.</i>	Sistema de Ventas de la Tienda Importaciones Laban S.A.C - Huancabamba .....	45
<i>Figura 9.</i>	Logo Kunagi.....	60
<i>Figura 10.</i>	Logo ScrumDo .....	61
<i>Figura 11.</i>	Logo Sprintometer .....	61
<i>Figura 12.</i>	Logo iceScrum .....	62
<i>Figura 13.</i>	Logo Pango Scrum .....	63
<i>Figura 14.</i>	ScrumDo .....	64
<i>Figura 15.</i>	Backlog .....	64
<i>Figura 16.</i>	Backlog 1 Historias de Usuario.....	65
<i>Figura 17.</i>	Backlog 2 From de creación de Historia de Usuario .....	66
<i>Figura 18.</i>	Detalle de Historia.....	67
<i>Figura 19.</i>	Menú de Rótulo.....	68
<i>Figura 20.</i>	Estado Historia .....	68
<i>Figura 21.</i>	Puntos de Historia .....	69
<i>Figura 22.</i>	Tiempo de Historia.....	69
<i>Figura 23.</i>	Assignees Team .....	70
<i>Figura 24.</i>	Tags de Historia de Usuario .....	70
<i>Figura 25.</i>	Navigation 1 Menú de Navigation (Navegación) .....	71
<i>Figura 26.</i>	Board 1 Pizarra de Trabajo .....	72
<i>Figura 27.</i>	Reportes de Avance del Sprint.....	73
<i>Figura 28.</i>	Menú Principal .....	74
<i>Figura 29.</i>	Marcador1 .....	76
<i>Figura 30.</i>	Marcadores .....	77



Figura 31.	Objeto 3D Cama.....	78
Figura 32.	Objeto 3D Escritorio .....	78
Figura 33.	Objeto 3D Laptop Samsung .....	79
Figura 34.	Estructura de la Aplicación .....	80
Figura 35.	Archivo AndroidManifest.xml .....	82
Figura 36.	Estructura del proyecto con clases de cámara y animación 3d .....	83
Figura 37.	Asignando Objeto 3D al Marcador .....	91
Figura 38.	Lista de Objetos 3D.....	92
Figura 39.	Exportación de la aplicación a extension .apk .....	93
Figura 40.	Marcador1 impreso .....	94
Figura 41.	Marcador 2 Impreso .....	95
Figura 42.	Marcador 3 Impreso .....	95
Figura 43.	Marcadores impresos .....	96
Figura 44.	Scaneo de Prueba en marcador 1 .....	97
Figura 45.	Scaneo de Prueba en marcador 1 en proceso .....	98
Figura 46.	Muestra objeto 3D con marcador .....	99
Figura 47.	Muestra objeto 3D con marcador 1 .....	100
Figura 48.	Ventas de los meses de enero, febrero y marzo. ....	101
Figura 49.	Número de ventas por mes [enero –abril] .....	102
Figura 50.	Ventas por mes incluye [Enero-Abril] .....	103
Figura 51.	Encuesta a usuarios del mes de Abril.....	105

### Índice de Tablas:

Tabla 1.	Requisitos Priorizados Fuente: Elaboración Propia .....	56
Tabla 2.	Características de Smartphone Samsung Galaxy .....	59
Tabla 3.	Características de Smartphone Samsung Galaxy .....	59





## **Dedicatoria**

A mis padres y hermanos, por su apoyo incondicional, por su comprensión, amor y sabios consejos en todo momento.

A esas personas tan importantes que Dios me dió la dicha de conocer,

Mis amigos.



## **Agradecimiento**

A Dios por siempre estar conmigo y darme la fortaleza para poder realizar mi tesis de la mejor manera.

A mis padres por siempre estar acompañándome, aconsejándome y brindándome su apoyo incondicional.

A mis amigos que de alguna manera me brindaron su apoyo y ayuda para la elaboración de esta tesis.

A todas las personas que me brindaron su apoyo y cariño para poder llevar a cabo la elaboración satisfactoria de esta tesis.



## RESUMEN

En la presente tesis se realizará el diseño e implementación de una aplicación cuyo objetivo es servir dentro de la tienda Importaciones Laban S.A.C. para la toma de decisiones más rápida en los clientes a la hora de elegir un producto de la tienda y así mejorar las ventas en la empresa haciendo uso de la tecnología de Realidad Aumentada para mostrar imágenes en 3D de determinados productos. Para ello se plantea desarrollar una aplicación de Realidad Aumentada basada en marcadores, que puedan ser ubicados en folletos o catálogos, de tal manera que al ser reconocidos por la aplicación muestren una imagen en 3D del objeto que se quiera mostrar al posible comprador. Del mismo modo para complementar el sistema se incluirá una galería de los objetos que estén disponibles en la aplicación para poder ser visualizados. Esta tesis está estructurada de la siguiente manera:

En el primer capítulo se centra en mencionar el marco referencial, dentro de éste se muestra la descripción de la realidad problemática, la formulación del problema, el objetivo general como los específicos, el marco institucional de la Empresa, el marco teórico donde se menciona lo que son sistemas operativos para dispositivos móviles y todo lo relacionado con Realidad Aumentada, definición, elementos, tipos y herramientas de desarrollo, así como la evolución del campo de la Realidad Aumentada; luego se detalla la situación actual de los participantes en el negocio de los dispositivos móviles, las mejoras que existen hoy en día con Realidad Aumentada y el potencial mercado al cual apuntan y la hipótesis de la Investigación.

En el Segundo capítulo, se muestra el análisis de la empresa, tanto el área de ventas así como también el Análisis del sistema de Ventas. En el Tercer capítulo, analizamos la base de parámetros de los sistemas de realidad aumentada a Utilizar, también se describe la funcionalidad de como activar la realidad aumentada en los dispositivos móviles. En el Cuarto capítulo se describe la planificación de desarrollo de la aplicación con la metodología elegida en el capítulo Cuatro. En el Quinto capítulo se describe la implementación de la aplicación, el desarrollo y las pruebas de funcionamiento de la aplicación, y finalmente se realiza la constatación de resultados con la hipótesis planteada.



## ABSTRACT

In this thesis the design and implementation of an application will be made which aims to provide in-store Laban SAC Imports for making faster decisions on customers when choosing a product from the store and improve sales in the company using Augmented Reality technology to display 3D images of certain products. To this end it is proposed to develop an application based on augmented reality markers that can be placed in brochures or catalogs, so as to be recognized by the application displaying a 3D image of the object you want to show the prospective buyer. Similarly to complement the gallery system objects that are available in the application to be displayed is included. This thesis is structured as follows:

The first chapter focuses on mentioning the reference framework within it the description of the problematic reality shows, the problem formulation, the general objective and specific, the institutional framework of the Company, the theoretical framework mentioned it they are operating systems for mobile devices and everything related to Augmented Reality, definition, elements, types and development tools, as well as the evolution of the field of Augmented Reality; then the current situation of the participants in the business of mobile devices, improvements that exist today with augmented reality and potential market that aim and the hypothesis of the research is detailed.

In the second chapter, analysis of the company, both the sales and also the sales analysis system is shown. In the third chapter, we analyze the parameters based augmented reality systems to use, functionality and enable augmented reality on mobile devices is also described. In the fourth chapter planning application development with the chosen methodology in Chapter Four described. In the fifth chapter the application deployment, development and operational testing of the application is described, and finally the verification of results is done with the hypothesis.



## INTRODUCCIÓN:

Hoy en el mundo los negocios se manejan de muchas maneras y para esto las empresas grandes siempre están en la búsqueda de mejores estrategias de convencimiento al público con respecto a sus productos, y año tras año la forma de marketing está evolucionando, tanto es la competencia en todos los campos que el uso de las tecnologías es parte fundamental para poder llevar acabo esta labor; los servicios y artículos tecnológicos, han cambiado tanto que ahora ya no se usan muchos de ellos o ya no tienen el mismo impacto como lo tenían años atrás como por ejemplo los periódicos o carteles estáticos.

Hoy en día casi todo está digitalizado, cuando salimos por las calles y centros comerciales y vemos letreros enormes con anuncios muy distintos, cambiantes y actualizados, en la red vemos anuncios siempre con videos sobre determinados productos mostrando siempre lo que resalta en cada uno de ellos, ahora y ya desde hace algunos años en Europa y recientemente en América del Norte se está utilizando la Realidad Aumentada, esta tecnología está llegando con un impacto tremendo en todos los campos con los cuales se ha ido utilizando ya que tiene muchas ventajas y le da a los usuarios una sensación de realismo digital que simplemente impacta en toda persona que lo utiliza.

En el campo de las ventas la tecnología de la Realidad Aumentada no se hace ausente pues muchas empresas ya lo utilizan para el márketing y promoción de sus productos logrando incrementar sus ventas, esta tecnología lo que genera en los clientes es, que lo ayuda a la hora de tomar decisiones ya que muestra la realidad digital combinada con la real justamente con los productos que este está interesado en adquirir.

Es por ello que en nuestro país ya se está utilizando recientemente desde hace 2 años atrás y ahora la voy a utilizar esta tecnología en la empresa Importaciones Laban S.A.C – Huancabamba para ayudar a mejorar la toma de decisiones de los clientes y así mejorar las ventas de la empresa con la aplicación a desarrollar con esta tecnología de Realidad Aumentada.



## **Capítulo I MARCO REFERENCIAL**

### **I.1 PRESENTACIÓN DEL PROBLEMA**

#### **I.1.1 Descripción de la realidad problemática**

La publicidad es una técnica de comunicación comercial que intenta fomentar el consumo de un producto o servicio a través de los medios de comunicación.

En primer lugar, la publicidad informa al consumidor sobre los beneficios de un determinado producto o servicio, resaltando la diferenciación por sobre otras marcas.

En segundo lugar, la publicidad busca inclinar la balanza motivacional del sujeto hacia el producto anunciado por medios psicológicos, de manera que la probabilidad de que el objeto o servicio anunciado sea adquirido por el consumidor se haga más alta gracias al anuncio.

Asimismo el marketing, es la orientación con la que se administra el mercadeo o la comercialización dentro de una organización, busca la fidelidad de los clientes mediante herramientas y estrategias; posiciona en la mente del consumidor, un producto, marca, etc. buscando ser la opción principal y llegar al usuario final; parte de las necesidades del cliente o consumidor, para diseñar, organizar, ejecutar y controlar la función comercializadora o mercadeo de la organización.



Como se está conociendo cada vez más sobre esta herramienta tecnológica varias empresas Publicitarias y de Software en el mundo están empezando a darle un valor muy importante dentro de lo que es la publicidad y Marketing para las empresas y los resultados lo demuestran, para dar una idea más clara sobre lo que es la Realidad Aumentada. La realidad aumentada es la tecnología que permite añadir información e imágenes sobreimpresas a la realidad que vemos a través de dispositivos tecnológicos. Al contrario que la realidad virtual, en este caso no se trata de crear una nueva realidad, sino de enriquecer la existente.

La Realidad Aumentada (RA) también ofrece ventajas y nuevas oportunidades para el marketing. Las campañas apuestan cada vez más por las redes sociales, nuevas tecnologías móviles, donde la RA o dispositivos como las Google Glass permiten mejorar la experiencia del usuario y aumentar el impacto sobre el consumidor.

Es por eso que en muchos países Europeos y en algunos americanos ya están teniendo un gran auge para la promoción y marketing de las empresas por que está dando buenos resultados, esto nos dice que es una tecnología muy importante para el futuro del marketing de las empresas.



La empresa Importaciones Laban SAC es una empresa creada en Moyobamba y ahora cuenta con una sucursal en Huancabamba – Piura desde hace 1 año, con muchos años de experiencia dedicada principalmente a la venta de artículos para el hogar, como muebles, línea blanca, electrodomésticos, accesorios, celulares y también motocicletas; esta empresa se encuentra en un crecimiento moderado, la competencia siempre está tratando de ser más conocida con mejoras publicitarias o constantes propagandas. En esta empresa Importaciones Laban SAC – Huancabamba no se tiene mucha muestra de imagen ni promoción de sus productos con tecnología moderna de última generación, lo cual hace referencia que se podrían mejorar las ventas con una mejora en su sistema de ventas, si bien es cierto ahora también nos enfrentamos a una realidad tecnológica que constantemente está innovando como lo es inicialmente en nuestro país la Realidad Aumentada; tendríamos que invertir en una nueva tecnología para mejorar nuestro sistema de ventas y así mejorar las ventas de los productos de la tienda con miras a enfrentar las nuevas formas de competencia y así estar a la vanguardia con las empresas competidoras y ser tendencia en el norte de Nuestro País.

### **I.1.2 Formulación del Problema**

**¿De qué manera repercutirá en las ventas la creación de una aplicación de Realidad Aumentada para la publicidad en la empresa “IMPORTACIONES LABAN SAC-HUANCABAMBA”?**





## **I.2 OBJETIVOS**

### **I.2.1 Objetivo General.**

Implementar una Aplicación basada en Realidad Aumentada para mejorar las ventas de la empresa “IMPORTACIONES LABAN SAC-HUANCABAMBA”.

### **I.2.2 Objetivos Específicos.**

- Analizar la información de Marketing que maneja la empresa.
- Construir la base de parámetros, sobre la cual se podrá visualizar la Animación.
- Realizar el desarrollo de la Aplicación de Realidad Aumentada utilizando un framework de realidad aumentada.
- Realizar la encriptación y codificación de la animación en código QR e Imagen.
- Elaborar las pruebas correspondientes de la aplicación de Realidad Aumentada con la animación previamente construida y realizar la puesta en marcha de la aplicación con Realidad Aumentada.



### **I.3 MARCO INSTITUCIONAL**

La empresa IMPORTACIONES LABAN SAC inició en el año 2001 en la ciudad de Moyobamba departamento de San Martín, esta empresa se dedica especialmente a lo que es venta de electrodomésticos, muebles para el hogar y también motocicletas de todo tipo. Esta empresa ha ido creciendo con el pasar del tiempo y ahora ya cuenta con 5 tiendas en el mismo departamento y con tiendas en otros departamentos incluidos Huancabamba-Piura. La expectativa de la tienda es crecer a nivel nacional e internacional y para esto vienen trabajando de manera muy responsable y comprometida para lograr su objetivo.

### **I.4 MARCO TEORICO**

Es conveniente conocer previamente sobre los sistemas operativos para móviles ya que se utilizará una tecnología que recién se está tomando en cuenta en el mercado como es la realidad aumentada, luego se conocerán sus características y las herramientas de desarrollo de esta tecnología que permitan agregar esta tecnología en los dispositivos móviles.



## **Sistemas Operativos Móviles**

En la actualidad hay diferentes opciones en el mercado en cuanto a sistemas operativos móviles se refiere. Los sistemas operativos son un poco más simples que los sistemas operativos que se utilizan en las computadoras hoy en día (Windows, Linux, Mac Os, entre otros), pero en esencia son lo mismo. Un sistema operativo se encarga de controlar el hardware del dispositivo para así brindar soporte a otras aplicaciones que se ejecutan en el mismo. A diferencia de los sistemas operativos para computadoras, los sistemas operativos móviles están orientados a la conexión inalámbrica, formatos multimedia comunes en estos dispositivos y métodos de entrada de datos para el hardware específico (pantallas táctiles, teclados, botones, etc). Es un conjunto de programas que gestiona los recursos de hardware y provee servicios en un dispositivo móvil, dicho de otra manera se podría decir que es un gran programa hecho para controlar los otros programas o aplicaciones. También controla las partes físicas de un dispositivo, por lo tanto, sin un sistema operativo no se podrían tomar fotos, capturar video, usar el teclado o ejecutar aplicaciones.

Es importante destacar que no todos los sistemas operativos son iguales, por lo que un programa que corre en un sistema operativo específico, probablemente no funcionará en otro. Es muy importante conocer las características de los diferentes sistemas operativos para evaluar si se ajusta o no a nuestras necesidades. Ejemplos de sistemas operativos para celulares son: Android, Windows Phone, IOS, Symbian y otros desarrollados específicamente para una u otra marca de celulares.

### **Android**

Android es un sistema operativo orientado a dispositivos móviles, basado en una versión modificada del núcleo Linux. Inicialmente fue desarrollado por Android Inc., una pequeña empresa, que posteriormente fue comprada por Google; en la actualidad lo desarrollan los miembros de la Open Handset Alliance (liderada por Google). (Orozco, 2011)



## **iOS**

iOS es un sistema operativo móvil de Apple. Originalmente desarrollado para el iPhone, siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV. Apple, Inc. no permite la instalación de iOS en hardware de terceros. La interfaz de usuario de iOS está basada en el concepto de manipulación directa, usando gestos multitáctiles. Los elementos de control consisten de deslizadores, interruptores y botones. La interacción con el sistema operativo incluye gestos como deslices, toques, pellizcos, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz. (Pedro Gutiérrez, 2011)

## **Windows Phone OS**

La compañía reinventó la idea de la compatibilidad entre plataformas con el lanzamiento de Windows Phone 8 a finales del año 2012. Aprovechando el lanzamiento oficial de Windows 8 para PC, la nueva plataforma móvil está diseñada para ofrecer a los usuarios una transición sin problemas desde el entorno de escritorio. Además de la liberación de muchos modelos de teléfono sólidos fabricados por HTC y Nokia (principalmente), Microsoft también introdujo una nueva línea de tabletas con el software actualizado. En teoría, un usuario puede acceder a sus archivos importantes en cualquier lugar gracias a las capacidades multi-plataforma de Windows 8. (Dubretic, 2014)

## **Symbian**

Symbian es un sistema operativo que fue producto de la alianza de varias empresas de telefonía móvil, entre las que se encuentran Nokia, Psion, Samsung, Arima, Benq, Fujitsu, Lenovo, LG, Motorola, etc. Sus orígenes provienen de su antepasado EPOC 32, utilizado en PDA's y Handhelds de PSION. El objetivo de Symbian fue crear un sistema operativo para terminales que pudiera competir con el plan de Palm o el Windows Phone de Microsoft y ahora Android de Google Inc. Y iOS de Apple Inc.



## Análisis de los Sistemas Operativos Móviles

### Análisis de Mercado

Para saber a qué sistema operativo va dirigida nuestra aplicación tenemos que analizar el público objetivo y los requerimientos técnicos del sistema operativo. Las ventas de smartphones a nivel mundial han crecido un 27%, llegando a los 295 millones de unidades en el segundo trimestre de 2014. Según un informe de la unidad WSS (Wireless Smartphone Strategies) de Strategy Analytics, Android representa un 85% de esas ventas globales. Se trata de un nuevo récord de cuota de mercado para Android, que se mantiene por delante de otros sistemas operativos. (Javier Guadiana, 2014)

Así quedaría la clasificación global según la cuota de mercado de cada uno de ellos:

Global Smartphone OS Shipments (Millions of Units)	Q2 '13	Q2 '14
Android	186.8	249.6
Apple iOS	31.2	35.2
Microsoft	8.9	8.0
Blackberry	5.7	1.9
Others	0.5	0.5
<b>Total</b>	<b>233.0</b>	<b>295.2</b>

Global Smartphone OS Marketshare %	Q2 '13	Q2 '14
Android	80.2%	84.6%
Apple iOS	13.4%	11.9%
Microsoft	3.8%	2.7%
Blackberry	2.4%	0.6%
Others	0.2%	0.2%
<b>Total</b>	<b>100.0%</b>	<b>100.0%</b>

Total Growth Year-over-Year %	48.9%	26.7%
-------------------------------	-------	-------

**Figura 1.** Sistemas Operativos en el Mercado

**Fuente:** (Javier Guadiana, 2014)



## **Análisis Técnico**

Apple presentó a bombo y platillo sus iPhone 6 y 6 Plus. Con ellos ha llegado una nueva versión del sistema operativo. iOS 8 se adelantó a Android 5.0 Lollipop por semanas, aunque de la nueva versión de Android ya se conocen muchas de sus funciones antes de que sean publicadas oficialmente en su lanzamiento. Muchos usuarios piensan que el sistema operativo de Apple es el más robusto y seguro, además, en la era en la que estamos las ideas se deslizan muy fácilmente entre un sistema y otro. Comparación entre iOS 8 y Android 5.0 Lollipop. A continuación el análisis de estos sistemas operativos. (Isabel Valencia, 2014)

### **Diseño**

El gran cambio de iOS llegó con la versión iOS 7, donde Apple dejaba atrás su skeuomorfismo para presentar unos iconos más planos y coloridos, no al gusto de todos. El cambio a iOS 8 no ha sido tan dramático, sino más bien lo podemos considerar como un refinamiento. En él encontramos alguna de las características que ya se dieron a conocer en el WWDC, tales como sugerencia de palabras automáticas, la posibilidad de utilizar teclados de terceros, la posibilidad de compartir archivos al estilo Google Drive y la posibilidad de que las aplicaciones envíen información entre ellas.

Por su parte, Android también ha optado por iconos planos, aunque este diseño, introducido con Android KitKat está sufriendo un cambio hacia lo que Google ha denominado Material Design. El menú de ajustes y el cajón de aplicaciones pasan a tener un fondo blanco con tono pastel, al contrario del negro que solía tener con KitKat.

La barra de notificaciones ha sufrido ciertos cambios. Así, las notificaciones y ajustes rápidos se reúnen en una única barra, haciendo que dé igual si arrastramos uno o dos dedos. Al abrir la barra de notificaciones veremos primero las notificaciones, si deslizamos otra vez, aparecen los ajustes rápidos. Desde allí tendremos acceso directo a los ajustes de cada uno como os contamos en estos trucos de Lollipop. (Isabel Valencia, 2014)



### **Rendimiento**

Al igual que su predecesor, **iOS 8 es un sistema operativo que soporta 64 bits**, mientras que en la esfera Android, la versión nueva será, por fin, totalmente compatible con procesadores de 64 bits, a lo que añadimos un tiempo nuevo de ejecución. Android Lollipop deja de lado dalvik en favor de ART, que nos promete una mejora en el rendimiento de aplicaciones, lo que quiere decir mayor rendimiento general.

En Android Lollipop podemos esperar algo de lag en ocasiones puntuales, algo que rara vez encontraremos en iOS. Esto es debido a que no está 100% optimizado para todos los terminales. Me explico, iOS es un sistema operativo que únicamente encontramos en iPad, iPod e iPhone, por tanto, solo tiene que ser optimizado para 3 terminales. Android va a incluir Lollipop en muchísimos terminales, y no es fácil optimizarlo para todos. Es bien sabido por todos que no es lo mismo un Galaxy S5, un Note 4, un Nexus 5 o un LG G3. Cada uno tiene su peculiaridad, su interfaz y su hardware específico. (Isabel Valencia, 2014)

### **Personalización**

Apple nos ha mostrado que está dando sus primeros pasos hacia más flexibilidad en cuanto a personalización se refiere, aunque todavía los usuarios de Android son más libres y pueden customizar más sus dispositivos. Más allá de los fondos de pantalla y tonos de llamada, alguien que tenga un Android podrá cambiar el lanzador, la pantalla de bloqueo y muchas cosas más.

Si hablamos del mundo root las posibilidades son ya casi infinitas, aunque también depende del terminal. Si tienes un móvil muy vendido, la cantidad de desarrolladores que trabajen en ellos será mayor, y podrás aprovecharte de una buena cantidad de ROMs diferentes, de aplicaciones de personalización, de MODs de diferentes aplicaciones y más cosas. (Isabel Valencia, 2014)



## **Batería**

Con Android Lollipop ha llegado una nueva forma de ahorro de batería, lo que en un principio conocimos como Project Volta. La nueva tecnología promete el aumento de hasta un tercio en la autonomía de nuestra batería. Además, gracias a ART también veremos una mejora de la vida de nuestra batería. De momento lo hemos podido probar poco, y el sabor de boca que se nos ha quedado es amargo. No, no funciona como nos prometían (y así pudimos comprobar en el análisis del Nexus 9), pero creemos que en una próxima actualización de hardware este problema se verá solucionado total o parcialmente.

En cuanto a Apple, con iOS 8 se habla de una batería más eficiente, sin embargo, esto en realidad solo se apreció con el iPhone 6 Plus, que sí tiene una batería suficiente. El iPhone 6 de menor tamaño sufre los problemas de su antecesor, el iPhone 5S. La batería desaparece en un abrir y cerrar de ojos. (Isabel Valencia, 2014)

## **Juegos**

Apple tiene Metal, una nueva API para desarrolladores de juegos y Android Lollipop tiene OpenGL ES 3.1, así como Android Extension Pack. Los dos muy potentes para juegos con gráficos de calidad y capaces de convertir tu smartphone en una auténtica videoconsola.

## **Actualizaciones**

Una gran ventaja de los dispositivos de Apple es que las actualizaciones suelen ser muy puntuales y llegan a todos los terminales a la vez, algo que no suele ocurrir con los dispositivos Android. Al mes de estar iOS disponible, ya había llegado al 50% de los teléfonos aproximadamente, algo que Lollipop no puede ni soñar.





Tenemos que admitir que Google va poco a poco por el buen camino gracias a Play Services que nos garantiza actualizaciones cada seis semanas para prevenir que algunos dispositivos se queden atrás. El tiempo que hay y habrá entre la llegada al mercado del Nexus 9 y el proceso de actualización de terminales es ligeramente más corta que la que experimentamos con KitKat.

La tendencia está marcada por la salud. Por eso, ambos sistemas presentan plataformas de salud para smartphones y otros dispositivos como los wearables. Apple tiene su Health App, en la que trabaja junto a Mayo Clinic y que promete ser una forma segura de almacenar datos, sin compartirlos con terceros. Para los desarrolladores se encuentra la API llamada HealthKit. (Isabel Valencia, 2014)

### **Ecosistema de Android Lollipop vs iOS 8**

Los dos sistemas operativos están presentando una sana evolución. Ambos comenzaron exclusivos para smartphones y se han ido abriendo camino en tablets y, actualmente, en los wearables. El enfoque es el mismo, tanto iOS 8 como Android Lollipop pretenden crear un ecosistema donde se puedan aunar tres grandes tipos de tecnología: domótica, registro de salud y los sistemas de automóvil. Veremos qué tal van acercándose a esos caminos.

Android sigue estando por años luz por delante en términos de personalización, aunque Apple parece mostrar una casi perfecta integración de hardware y software. Por otro lado, Apple presenta una tienda de aplicaciones más ordenada, pero obliga a sus usuarios a mantener alguna de sus aplicaciones. Una de las ventajas del sistema operativo Android es que el usuario que lo elija se va a encontrar con una amplia gama de smartphones donde elegir, así como la opción de eliminar aplicaciones nativas o de la operadora, cosa que en iOS no se puede hacer. (Isabel Valencia, 2014)



## **Realidad Aumentada**

La realidad aumentada surgió por primera vez en los años 70, como una tecnología orientada a las experiencias en mundos virtuales. El término fue acuñado por Tom Caudell en 1992, y a partir de ese momento se sucedieron diferentes aplicaciones y plataformas para desarrollar más tecnología y aplicaciones de realidad aumentada.

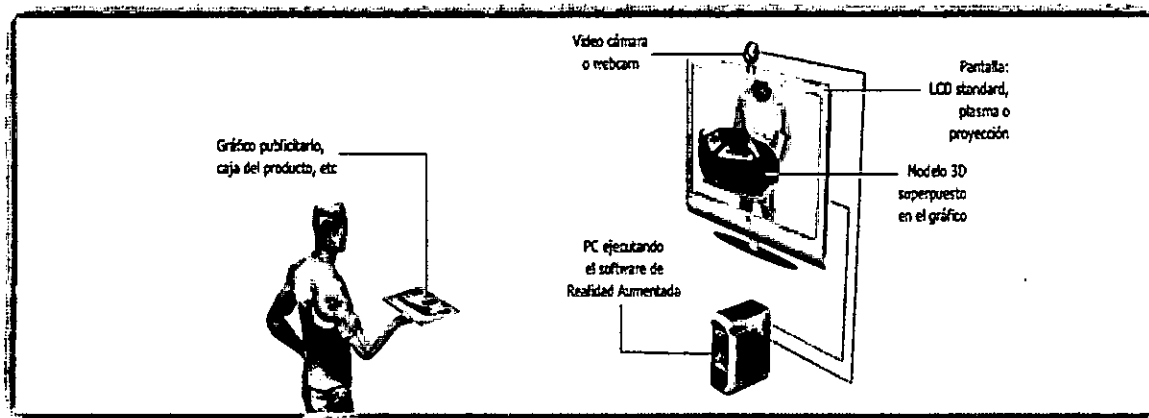
Ya en el siglo XXI la realidad aumentada ha entrado en un periodo de auge, que se ha dividido en 3 etapas:

### **Realidad Aumentada en ordenadores personales.**

Entre el año 2006 y 2008, gracias al mundo de los videos juegos y a la mejora de las capacidades computacionales de ordenadores y tarjetas gráficas, resultó posible confeccionar experiencias de realidad aumentada de una gran calidad. Las máquinas eran capaces de mover escenas tridimensionales de más de 100.000 polígonos al mismo tiempo que se realizaba el tracking de los elementos visuales. Fueron muy populares en esos años las aplicaciones de marketing, tanto en punto de venta y eventos on stage, como integradas en páginas Web. Aparecieron en el mercado las primeras herramientas de programación de realidad aumentada de alto nivel (D'Fusion de Total Immersion o Metaio SDK) y proliferaron las empresas especializadas en este campo. (Innovae, 2013)

La realidad aumentada consiste en combinar el mundo real con el virtual mediante un proceso informático, enriqueciendo la experiencia visual y mejorando la calidad de comunicación.

Gracias a esta tecnología se puede añadir información visual a la realidad, y crear todo tipo de experiencias interactivas: Catálogos de productos en 3D, probadores de ropa virtual, video juegos y mucho más.



**Figura 2. Arquitectura de Realidad Aumentada 1**

**Fuente: (Salazar Alvares, 2013)**

En el gráfico se puede apreciar el proceso informático que sucede en los sistemas de realidad aumentada. Normalmente se requiere una cámara de vídeo, un monitor y un ordenador con un software especial instalado. La realidad aumentada es ya muy popular en Apps que funcionan en todo tipo de teléfonos inteligentes (smartphones) y Tablets. (Innovae, 2013)

### **Realidad Aumentada en smartphones.**

La revolución social y tecnológica provocada unos años más tarde por el visionario Steve Jobs con la invención de los smartphones, actuó como catalizador en el desarrollo de la realidad aumentada. Los smartphones y posteriormente también las tabletas, permitían a los usuarios disfrutar de las experiencias de realidad aumentada de forma inmediata. Proliferaron aplicaciones vinculadas a revistas, catálogos o carteles publicitarios. También aparecieron las primeras Apps en el sector turístico que vinculaban información de internet a una capa superpuesta a la cámara del móvil, en función de la orientación y la localización de un usuario (Wikitude o Layar). (Innovae, 2013)



**Figura 3. Ejemplo Realidad Aumentada-Tablet**

**Fuente: Elaboración Propia**

#### **Realidad Aumentada en gafas y visores.**

En la actualidad estamos viviendo la siguiente revolución de la tecnología de realidad aumentada gracias al empujón mediático que Google propició para sus gafas de realidad aumentada. Aunque todavía los dispositivos que existen en el mercado son un tanto toscos y la experiencia visual es muy mejorable, ya se intuyen numerosas aplicaciones y negocios en nuevos ámbitos como la formación profesional, la educación y el ocio digital. (Innovae, 2013)



**Figura 4. HoloLens**

**Fuente: Diario El comercio**

La realidad aumentada combina elementos virtuales con el mundo físico, permitiendo crear una realidad mixta multifuncional. Google Glass es un ejemplo de este tipo de tecnología, ya que permite al usuario interactuar con imágenes, textos y aplicaciones en un entorno no virtual.

Los HoloLens, a diferencia de otros dispositivos de realidad aumentada, prometen un sistema de hologramas nunca antes visto, es "la computadora holográfica más avanzada que el mundo haya conocido", señala Alex Kipman, el hombre detrás de este dispositivo.

Cuando hablamos de HoloLens, estamos hablando de tres tecnologías que trabajan en conjunto: Windows Holographic, que permite integrar imágenes 3D en todos los dispositivos en base a Windows 10. HoloLens, lentes que te otorgan la capacidad ver imágenes 3D e interactuar con ellas. Y HoloStudio, la aplicación mediante la cual podrás crear tus propios hologramas. (CNET, 2015)



## **Elementos del Sistema**

Esta tecnología de Realidad Aumentada necesita en esencia 4 elementos fundamentales para poder desarrollar el funcionamiento de la Realidad Aumentada, un elemento que captura las imágenes de la realidad, un elemento sobre el cual proyectar la mezcla de imágenes reales con las imágenes sintetizadas lo que sería una pantalla, un elemento de procesamiento y un activador de la información virtual.

### **Elemento Capturador**

Este elemento se encarga de captar la imagen del mundo real y transmitirla a la aplicación de Realidad Aumentada. Este elemento puede ser una cámara web, la cámara de un Smartphone, laptop o Tablet, pero esto va a depender de que características mínimas necesite en la aplicación para un óptimo funcionamiento de la aplicación.

### **Activación de Realidad Aumentada**

Este elemento sirve para alimentar a la aplicación de realidad aumentada he indicarle que información debe superponer sobre la realidad y complementarla. Existen varios elementos que pueden activar el proceso de Realidad Aumentada, y los menciono a continuación:

**Elementos de localización** como GPS, brújulas y acelerómetros, que permiten identificar la posición y orientación de los dispositivos en los que están integrados. En la actualidad estos elementos están presentes en gran parte de los dispositivos móviles. (M RUTH GAMERO, 2011)

**Marcador (símbolo impreso) o imagen**, sobre los que se superpone algún tipo de información previamente asignada (imagen, objetos 3D, vídeo,...) cuando son reconocidos por un software determinado. (KZGUNEA, 2013)

**Sensores** o cualquier otro elemento capaz de suministrar una información que represente lo que ven la persona usuaria. (KZGUNEA, 2013)



### **Elemento Procesador**

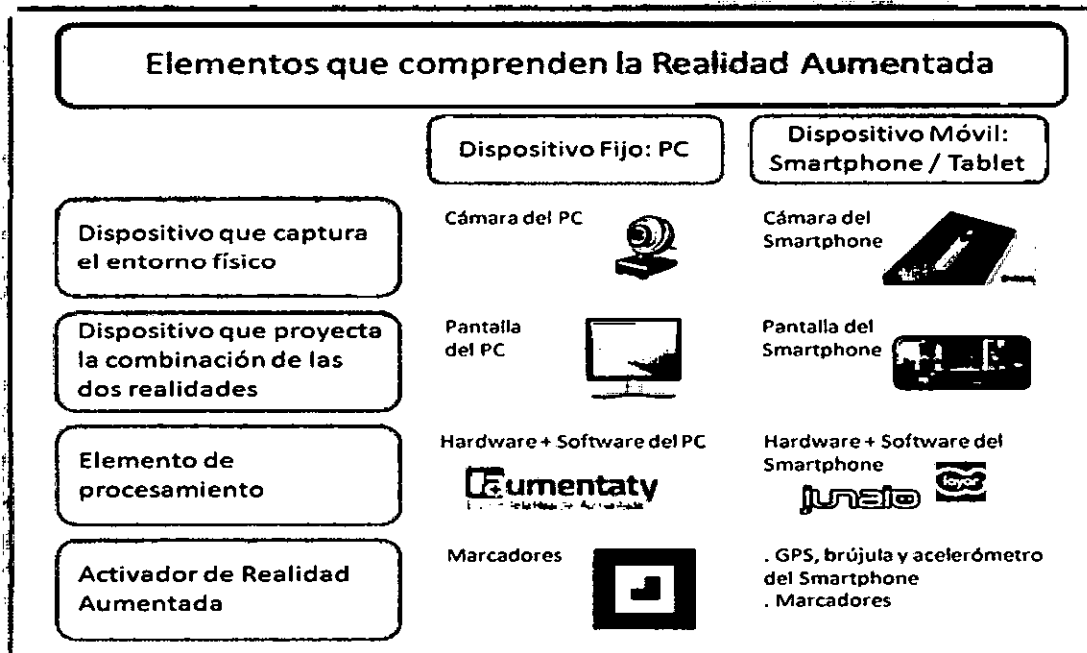
Este elemento es el software el cual se encarga de procesar la información que recibe del elemento capturador y del Activador de Realidad Aumentada, este elemento se encarga de procesar correctamente esta información y combinarla para que esta sea mostrada en el elemento sobre el cual se proyectará. Este software debe contar con un módulo de reconocimiento de imágenes, orientación espacial y superposición de imágenes para esto estará instalado en el dispositivo sobre el que se ejecute el proceso de Realidad Aumentada: PC, Smartphon o Tablet. (KZGUNEA, 2013)

### **Elemento sobre el cual proyectar**

Este elemento es muy necesario para poder visualizar la Realidad Aumentada y este puede ser la pantalla de un ordenador, de un Smartphone, Tablet. Acá se puede visualizar el funcionamiento de todos los elementos anteriores combinados y haciendo un trabajo en conjunto para por mostrar lo que es la Realidad Aumentada.



En resumen los elementos que comprenden la realidad son los siguientes:



**Figura 5. Elementos que Comprenden la Realidad Aumentada**

**Fuente: (KZGUNE, 2013)**

### Dispositivos para el desarrollo de Realidad Aumentada

Una vez conocidos los elementos necesarios para el funcionamiento de la Realidad aumentada podemos mencionar herramientas que sirvieron para el desarrollo de esta tecnología.





**Head-Mount Displays** Los HMD son dispositivos que se montan en la cabeza del usuario obligándolo a ver por una pantalla. Estos están conectados a una unidad de procesamiento, la cual envía la imagen al HMD y este la proyecta al usuario. Cuentan con una cámara que permite ver la perspectiva del usuario, así como detectores de movimiento que miden la posición y orientación de la cabeza. Podemos distinguir dos tipos de HMD: Opacos, estos encapsulan la cabeza del usuario, haciendo que este no pueda ver más allá de la pantalla del dispositivo. Semitransparentes, este tipo de HMD no obstruye completamente la visión del usuario, ya que cuenta con lentes semitransparentes, que permiten ver a través de ellos y a la misma vez mostrar imágenes virtuales que se superponen a las del entorno real. (Salazar Alvares, 2013)

**Head-Up Display:** Se considera un HUD cualquier objeto transparente que muestre información sobre él y además permita observar lo que hay detrás de este.

**Dispositivos móviles:** En el concepto de dispositivos móviles englobaremos tanto teléfonos móviles como tabletas. Estos son dispositivos que pueden ser transportados por el usuario fácilmente y cuentan con un procesador y una cámara. La cámara permitirá captar el escenario real, el procesador junto con los programas de Realidad Aumentada transformará esta información en imágenes reales y virtuales combinadas y la desplegarán sobre la pantalla del dispositivo. (Salazar Alvares, 2013)



## **Clasificación de sistemas de Realidad Aumentada**

Esta clasificación es tomada en cuenta según la forma de obtener la información, la cual permite la activación de la Realidad Aumentada.

### **Sistemas basados en el reconocimiento de marcadores**

Un marcador es un objeto cuya imagen es conocida por el sistema (Cawood et al. 2008; Choudary et al., 2009; Tateno, 2007). Dichos sistemas son reconocidos por usar marcadores fiduciales para obtener mediante una cámara y técnicas de visión por computador. Las maneras en que el sistema conoce el marcador se pueden agrupar en tres conjuntos, mediante su geometría, su color o mediante ambas características. Dentro de los sistemas de detección de marcas podemos destacar ARToolKit, ARTag, StudierStube, OsgART, etc. Estos sistemas están compuestos por una serie de librerías que recogen las técnicas de visión necesarias para realizar el posicionamiento del objeto. (Bautista Rojas & Archila Díaz, 2011)

### **Sistemas basados en geolocalización**

Este sistema se caracteriza porque es un tipo de Realidad Aumentada especialmente desarrollado para dispositivos móviles y que se sirve del propio hardware del dispositivo (GPS, brújula, acelerómetro, etc.) para geolocalizar la posición del usuario y mostrarle capas de información virtual sobre los diferentes Puntos de Interés que le rodean. Por medio del compás y el acelerómetro la orientación e inclinación del dispositivo; con esto sabe hacia el lugar que está apuntando la cámara. Las imágenes virtuales que se proyectan están basadas en coordenadas de tal manera que si el dispositivo está apuntando hacia estas coordenadas se mostrará la imagen virtual con información asociada. (Salazar Alvares, 2013)



## **Sistemas basados en el reconocimiento de formas**

Este sistema se basa en la comparación de imágenes o formas que se encuentran en la realidad captados con la cámara con los que tiene la aplicación guardados previamente en una base de datos, una vez que encuentra la similitud activa la animación 3D de la Realidad Aumentada. Este sistema a diferencia del basado en marcadores es más atractivo. (Salazar Alvares, 2013)

## **Herramientas de desarrollo para Realidad Aumentada**

Para llevar a cabo un buen desarrollo de aplicaciones móviles de Realidad Aumentada, uno de los puntos clave es contar con un software adecuado para desarrollarlo. Son varias las herramientas que nos pueden ayudar a desarrollar una app de estas características, pero algunas que se utilizan bastante en el sector del desarrollo de apps de RA son las siguientes. (Salazar Alvares, 2013)

### **Entorno de Desarrollo Integrado**

Un entorno de desarrollo integrado, llamado también **IDE** (sigla en inglés de integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación. Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). (Salazar Alvares, 2013)



**Editor de código fuente:** Editor de texto que sirve para editar el código fuente de aplicaciones informáticas.

**Un compilador:** Es un traductor de código fuente, lo traduce a un lenguaje que sea legible para las máquinas.

**Un depurador:** Es una aplicación que tiene como función probar y eliminar posibles errores en un programa en desarrollo.

**Constructor de interfaz gráfica:** Herramienta que sirve para crear y diseñar las interfaces con las cuales habrá interacción entre la aplicación y el usuario.



## Frameworks para Realidad Aumentada

- **Metaio:** Es la herramienta de realidad aumentada más utilizada en el desarrollo de aplicaciones móviles a nivel mundial. Cuenta con una cantidad enorme de herramientas para el desarrollo de RA. Está orientada especialmente a cuatro sectores: RA en impresión, marketing, industria y automoción. Destaca por su dilatada experiencia, de más de diez años, con grandes marcas como Mitsubishi, Lego, Audi o Ikea. (Salazar Alvares, 2013)

- **Layar:** A pesar de que no es una de las más potentes ni de las que más opciones ofrecen al desarrollador de aplicaciones móviles de Realidad Aumentada, sí que es de las más sencillas de utilizar. Su sencillez y facilidad de uso han logrado convertirla en una de las herramientas de desarrollo más conocidas. De hecho, con Layar han trabajado marcas como Ford, Reebok o Lonely Planet. Se centra sobre todo en la realidad aumentada a partir de proyectos impresos como revistas, carteles publicitarios o libros de texto, aunque aplicándolo a una cantidad asombrosa de sectores (desde publicidad hasta la educación). (Salazar Alvares, 2013)

- **Vuforia:** la plataforma que cuenta con el apoyo de Qualcomm es una herramienta muy buena para desarrollar cualquier tipo de experiencia de Realidad Aumentada sin importar el dispositivo o soporte en el que será ejecutada. Por ejemplo, permite perfectamente desarrollar aplicaciones móviles para iOS y para Android que utilicen Realidad Aumentada. (Salazar Alvares, 2013)

- **Total Immersion:** Esta es una de las plataformas más potentes del mercado. Además cuenta con herramientas de todo tipo para creación de experiencias en varias plataformas: Android, iOS, etc. Desde visualización de muebles, pruebas de prendas o realidad aumentada aplicada a las ventas y a marketing son algunos de los puntos fuertes de Total Inmersión. (Salazar Alvares, 2013)



- **ARTool Kit:** es la versión de software libre para el desarrollo de aplicaciones de Realidad Aumentada. Su mayor ventaja es que es una herramienta gratuita, a diferencia de las mencionadas anteriormente. También es libre **Mixare** que al igual que ARToolKit ofrece posibilidades muy interesantes pero que siguen siendo limitadas comparadas con las herramientas de pago. (Salazar Alvares, 2013)

- **VISION SDK:** Una librería para desarrolladores que permite incorporar de forma sencilla y rápida el innovador modelo de uso de RA a una App, ejecutando 'vistas' de RA de manera integrada con el resto de funcionalidades. De este modo, cualquier desarrollador puede incorporar la RA sin necesidad de invertir esfuerzos en desarrollar el código de programación específico, reduciendo sustancialmente el tiempo dedicado y asegurando la calidad de la App. (Salazar Alvares, 2013)

- **NyARToolkit:** Es una librería también basada en ARToolKit creada por Ryo Iizuka en 2008. En este caso no fue desarrollada específicamente para desarrollo aplicaciones Android, sino que el proyecto NyARToolkit es compatible con plataformas Java, C#, ActionScript 3, Silverlight 4, C++, Processing y obviamente Android, aunque no todos al mismo nivel en términos de estabilidad. De esta misma librería surgió otra, FLARToolKit, aún no disponible para Android.

NyARToolkit para Android ha sido desarrollada por un grupo de usuarios de Android en Japón y no existe apenas documentación online, y la que hay está escrita en japonés. Por eso, para entender esta librería hay que sumergirse directamente en el código fuente de Java para Android. Incluso la documentación del código está hecha en japonés, por lo que, y para que sirva de aviso al lector, a la hora de configurar el entorno de trabajo con esta librería es necesario editar las preferencias de codificación de todos los archivos a UTF-8, o no desaparecerán los errores. (Salazar Alvares, 2013)



## Análisis y Diseño

### Análisis de Herramientas para Realidad Aumentada

Ahora que ya he mencionado sobre las características de los sistemas operativos y los datos de mercado, tengo analizar las herramientas necesarias para desarrollar la aplicación de Realidad Aumentada. Sabiendo que estas herramientas dependen del tipo de sistema operativo que utilizaré.

	ARLAB	ARToolkit	DroidAR	Layar	Metaio	NyARToolkit	Vuforia
Reconocimiento de Marcas	✓	✓	✓	✗	✓	✓	✓
Múltiples marcas	✓	✓	✓	✗	✓	✓	✓
Geolocalización	✓	✗	✓	✓	✓	✗	✓
Reconocimiento de Formas	✓	✗	✗	✓	✓	✗	✓
Imágenes 3D	✓	✓	✓	✗	✓	✓	✓
Animación	✓	✓	✗	✓	✓	✓	✓
Android	✓	✗	✓	✓	✓	✓	✓
iOS	✓	✓	✗	✓	✓	✗	✓
Documentación	Alta	Alta	Media	Alta	Alta	Alta	Poca
Precio	Por Producto	Libre	Libre	Por Producto	Por Producto	Libre	Libre
Año de Publicación	2012	2010	2011	2009	2005	2008	2012

**Figura 6. Cuadro de Análisis de herramientas de desarrollo 3D**

**Fuente: Ivan Salazar pág. 41**

Como se observa Metaio es una muy buena opción de desarrollo, pero es un framework de pago y esto saldrá demasiado costoso pero también tenemos a NyARToolkit que es libre que sería a la que apuntaríamos.



## **Análisis de los requerimientos de la aplicación**

Es de vital importancia identificar los requerimientos de la aplicación que se desarrollará, ya que así se podrá comprender las condiciones y necesidades de la aplicación.

### **Requerimientos Funcionales**

- Reconocimiento de ubicación
- Visualización y manejo de imágenes 3D
- Visualización de imágenes
- Elección de imágenes 3D de una lista
- Visualización de varias imágenes 3D en el mismo contexto

### **Requerimientos No Funcionales**

- Fiabilidad
- Facilidad de Uso
- Optimada

### **Imágenes 3D**

Las imágenes que se mostrarán en la aplicación deben de ser de formato .md2, ya que este formato es el admitido por la librería NyARToolkit. Estas imágenes deben estar vinculadas a los productos que vende la empresa, como por ejemplos muebles para el hogar, electrodomésticos y motocicletas.

Las imágenes que usaré las he obtenido de Galería 3D - Sketchup – Google ya que encontré imágenes ya diseñadas para poder utilizarlas en la aplicación.





## **I.5 HIPOTESIS:**

### **I.5.1 Hipótesis de investigación**

Las ventas de la empresa Importaciones Labán SAC- Huancabamba mejorarán con la implementación de la Aplicación basada en Realidad Aumentada.

### **I.5.2 Operacionalización de las Variables**

- **VARIABLE INDEPENDIENTE:**

La aplicación basada en Realidad Aumentada.

INDICADOR DE LA VARIABLE INDEPENDIENTE

Creación y puesta en Marcha de la Aplicación

- **VARIABLE DEPENDIENTE:**

Las Ventas de la empresa Importaciones Laban SAC-Huancabamba.

INDICADOR DE LA VARIABLE DEPENDIENTE

Nivel de ventas en un determinado tiempo

### **Comprobación de la hipótesis**

Se probará la hipótesis comparando el promedio de ventas del mes de evaluación con los meses anteriores.



## Capítulo II ANÁLISIS DE LA EMPRESA

En este capítulo lo que se mencionará es la estructura del área de ventas de la empresa Importaciones Laban S.A.C. – Huancabamba, así como cada una de las funciones que cumple cada miembro que participa del área, de esta manera se pudo obtener una visión más realista y concisa con respecto al sistema de ventas de la empresa para así tener la perspectiva con la aplicación de realidad aumentada.

La empresa Importaciones Laban S.A.C está compuesta por varias áreas funcionales, dentro de las cuales están las diversas actividades más importantes ya que por ellas se plantean y tratan de alcanzar los objetivos y metas de la empresa. Generalmente una empresa está formada por lo menos de 5 áreas funcionales básicas (dirección, administración, mercado, ventas, producción, contabilidad y finanzas), pero puede estar formada por muchas más (investigación, recursos humanos, estrategia, etc). Como esta tienda con la cual se ha trabajado es una sucursal se ha enfocado directamente al área de ventas de la empresa.

El número de áreas funcionales en las pequeñas empresas se simplifican y se integran unas dentro de las otras.

- **Área de dirección:** área imprescindible, considerada la cabeza de la empresa está a cargo del señor Sergio Labán hermano mayor de los hermanos Labán el cual radica en la ciudad de Moyobamba y lleva el timón de la empresa, establece los objetivos y la dirige hacia ellos. Está relacionada con el resto de áreas funcionales, ya que es quien las controla.



- **Área de administración:** relacionada con el funcionamiento de la empresa a cargo del señor Amado Labán y que radica en la ciudad de Huancabamba. Encargado de la operación de negocio en sentido general, desde contrataciones, pagos a personal así como el control de todas las áreas dentro de la tienda en la ciudad de Huancabamba.

- **Área de Producción:** La empresa en esta área realiza todo lo concerniente al ensamblaje de motocicletas de dos y tres ruedas, así mismo del acabado y ensamblaje de artículos para el hogar como muebles y otros que posteriormente se pondrán a la venta en las tiendas, esta área cuenta con un aproximado de veinte trabajadores entre todas las áreas, diez en Moyobamba y diez en San Martín.

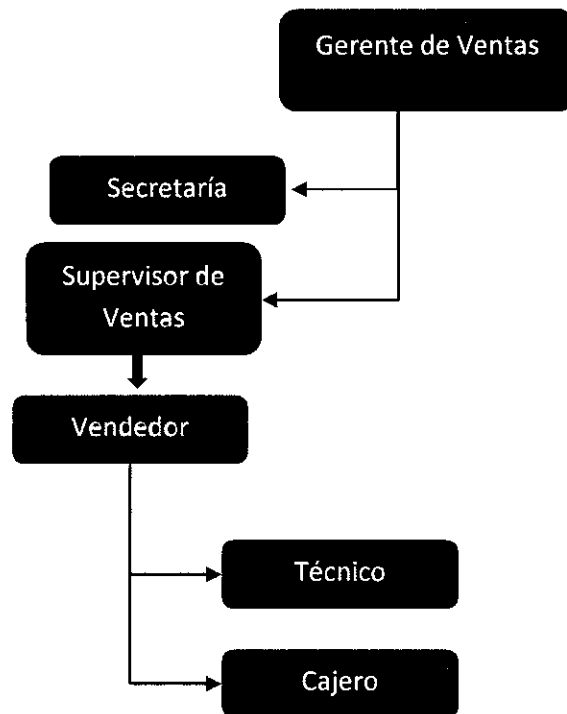
- **Área de contabilidad y finanzas:** Esta área está a cargo de dos trabajadores un Abogado encargado de ver toda la parte jurídica que esté vinculada a la empresa y un contador que tendrá en cuenta todos los movimientos de dinero, tanto dentro como fuera de la empresa.

- **Área de ventas:** orientada al exterior. En esta área se plantean las estrategias que la empresa seguirá en el área del marketing, los mercados donde la empresa opera, los segmentos de mercado, el ciclo de vida de los productos, diseño de nuevos productos. En la tienda sucursal ubicada en Huancabamba el área de ventas tiene la siguiente estructura.



## II.1 ÁREA DE VENTAS

La Empresa Importaciones Laban S.A.C- Huancabamba maneja su área de ventas de la siguiente manera:



**Figura 7.** Área Ventas de Importaciones Laban S.A.C – Huancabamba

**Fuente: Elaboración Propia**



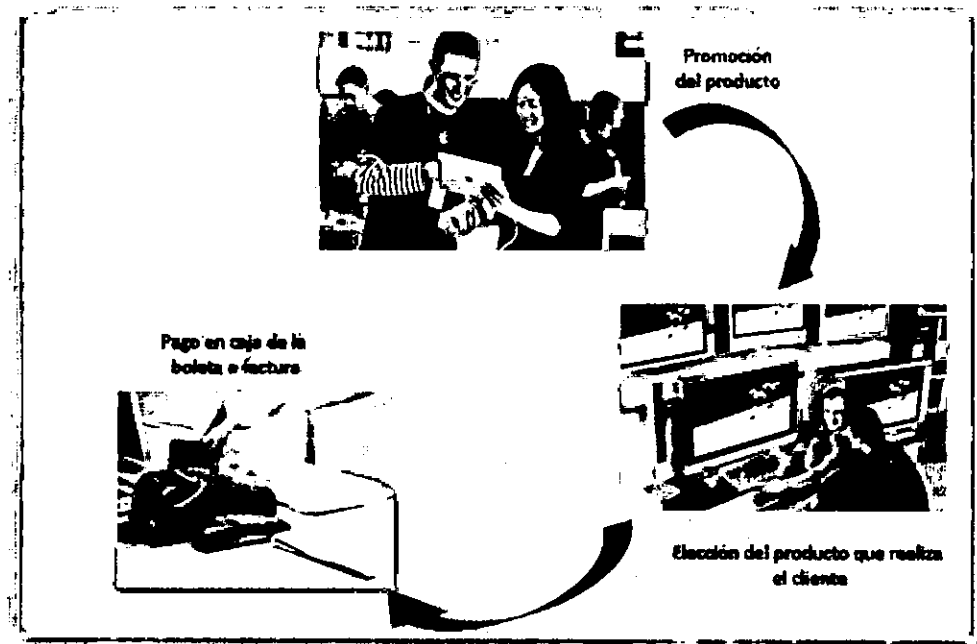
- El Gerente de ventas se encarga de manejar el control del funcionamiento de la tienda, propone los nuevos métodos de promoción de los productos y la forma de márketing que se utilizará en la tienda.
- La secretaria se encarga de llevar en regla y control todos los documentos, facturas y boletas de ventas de la tienda.
- El Supervisor encargado de supervisar y monitorear todas las ventas que se realizan en la tienda.
- El vendedor realiza la promoción directa de los productos a los posibles clientes, detallando las características de cada producto.
- El Técnico se encuentra disponible en la tienda para cuando se necesite la instalación de algún electrodoméstico que se haya comprado en la tienda o para servicio técnico que se algún cliente requiera.
- El cajero realiza el cobro de las boletas o facturas parte final de las ventas.

## II.2 ANÁLISIS DEL SISTEMA DE VENTAS

La empresa Importaciones Labán S.A.C sucursal Huancabamba maneja su sistema de ventas por medio de Transacciones comerciales con trato directo entre el comprador y el vendedor o venta asistida, dentro de esto la tradicional figura del dependiente se convierte en un «asesor comercial personal». La asesoría y la confianza son dos valores altamente apreciados por el consumidor, y es por eso que la empresa lo toma con una vital importancia en el comercio tradicional en donde encuentran su desarrollo en el acto de venta. Dentro del comercio tradicional se profesionaliza, se beneficia de las nuevas fórmulas y tecnologías que están disponibles en la localidad, dentro de estas están la propaganda por radio, publicidad en panel (Letrero grande) y su página web que está disponible en <http://www.laban.com.pe/>.



El sistema de venta se inicia desde la promoción del producto a los posibles clientes, luego es seguido por la elección de producto que desea comprar y finalmente este culmina con el pago en caja de la boleta o factura por el producto que el cliente ha elegido.



**Figura 8. Sistema de Ventas de la Tienda Importaciones Laban S.A.C - Huancabamba**

**Fuente : Elavoración Propia**

#### **Ventajas:**

- Pueden asociarse a una o varias centrales de compra (u otras fórmulas) sin asumir un coste elevado, ni perder su identidad.
- Cercanía con el cliente.
- Rápida respuesta y atención al cliente.



**Desventajas:**

- Falta de cultura de emprendedor.
- Escasa formación en gestión de ventas. No utiliza las modernas herramientas de merchandising, ni las nuevas tecnologías.
- Individualismo.



## **Capítulo III BASE DE PARÁMETROS**

En este capítulo se hace referencia a los tipos de sistemas de realidad aumentada, el sistema basado en reconocimiento de marcadores, el sistema basado en geolocalización y el sistema basado en el reconocimiento de formas. También se menciona lo concerniente a la activación de la realidad aumentada y sus elementos que lo componen para su funcionalidad dentro de la aplicación que se ha construido.

### **III.1 SISTEMAS DE REALIDAD AUMENTADA A UTILIZAR**

#### **Sistemas basados en el reconocimiento de marcadores**

Un marcador es un objeto cuya imagen es conocida por el sistema (Cawood et al. 2008; Choudary et al., 2009; Tateno, 2007). Dichos sistemas son reconocidos por usar marcadores fiduciales para obtener mediante una cámara y técnicas de visión por computador. Las maneras en que el sistema conoce el marcador se pueden agrupar en tres conjuntos, mediante su geometría, su color o mediante ambas características. Dentro de los sistemas de detección de marcas podemos destacar ARToolKit, ARTag, StudierStube, OsgART, etc. Estos sistemas están compuestos por una serie de librerías que recogen las técnicas de visión necesarias para realizar el posicionamiento del objeto. (Bautista Rojas & Archila Díaz, 2011) Como se está desarrollando dentro de la aplicación.





### **Sistemas basados en geolocalización**

Este sistema se caracteriza porque es un tipo de Realidad Aumentada especialmente desarrollado para dispositivos móviles y que se sirve del propio hardware del dispositivo (GPS, brújula, acelerómetro, etc.) para geolocalizar la posición del usuario y mostrarle capas de información virtual sobre los diferentes Puntos de Interés que le rodean. Por medio del compás y el acelerómetro la orientación e inclinación del dispositivo; con esto sabe hacia el lugar que está apuntando la cámara. Las imágenes virtuales que se proyectan están basadas en coordenadas de tal manera que si el dispositivo está apuntando hacia estas coordenadas se mostrará la imagen virtual con información asociada. (Salazar Alvares, 2013)

### **Sistemas basados en el reconocimiento de formas**

En este tipo de sistemas a diferencia de los basados en reconocimiento de marcas, ya no busca marcas determinadas sino formas conocidas. De igual manera lo captado por la cámara debe ser contrastado con una base de datos para tener una coincidencia de formas y poder mostrar la información asociada.



## III.2 ACTIVACIÓN DE REALIDAD AUMENTADA

### **Activación de Realidad Aumentada**

Este elemento sirve para alimentar a la aplicación de realidad aumentada he indicarle que información debe superponer sobre la realidad y complementarla. Existen varios elementos que pueden activar el proceso de Realidad Aumentada, y los menciono a continuación:

**Elementos de localización** como GPS, brújulas y acelerómetros, que permiten identificar la posición y orientación de los dispositivos en los que están integrados. En la actualidad estos elementos están presentes en gran parte de los dispositivos móviles. (M RUTH GAMERO, 2011)

**Marcador (símbolo impreso) o imagen**, sobre los que se superpone algún tipo de información previamente asignada (imagen, objetos 3D, vídeo,...) cuando son reconocidos por un software determinado. (KZGUNEА, 2013)

**Sensores** o cualquier otro elemento capaz de suministrar una información que represente lo que ven la persona usuaria. (KZGUNEА, 2013)

### **Elemento Procesador**

Este elemento es el software el cual se encarga de procesar la información que recibe del elemento capturador y del Activador de Realidad Aumentada, este elemento se encarga de procesar correctamente esta información y combinarla para que esta sea mostrada en el elemento sobre el cual se proyectará. Este software debe contar con un módulo de reconocimiento de imágenes, orientación espacial y superposición de imágenes para esto estará instalado en el dispositivo sobre el que se ejecute el proceso de Realidad Aumentada: PC, Smartphon o Tablet. (KZGUNEА, 2013)



### **Elemento sobre el cual proyectar**

Este elemento es muy necesario para poder visualizar la Realidad Aumentada y este puede ser la pantalla de un ordenador, de un Smartphone, Tablet. Acá se puede visualizar el funcionamiento de todos los elementos anteriores combinados y haciendo un trabajo en conjunto para por mostrar lo que es la Realidad Aumentada.

### **Elemento de Situación**

Son aquellos elementos que permiten posicionar la información virtual dentro de la realidad, por lo que cumplen una función importante dentro del sistema. Podemos clasificarlos en los siguientes elementos:

**GPS, brújula y acelerómetro:** Por medio del GPS se puede conocer la ubicación; con la brújula la dirección a la cual está apuntando el dispositivo y con el acelerómetro la inclinación. Se puede agregar información virtual basada en geolocalización, es decir si se apunta con el elemento capturado hacia una ubicación donde está definido un elemento virtual este se mostrara en la pantalla.

**Reconocimiento de objetos:** este método se basa en reconocer objetos conocidos como edificios o la forma de un objeto específico; para luego corroborarlo con una base de datos y mostrar la información virtual que se requiera.



### III.3 ELEMENTOS UTILIZADOS

Una vez que ya se ha visto los sistemas que se pueden utilizar, se realiza la elección de los sistemas que se utilizarán en la aplicación tomando en cuenta la factibilidad y facilidad de implementación y uso.

El primer sistema elegido es el Sistema de Reconocimiento de marcadores, este sistema se eligió por su fácil manejo y vinculación con cada objeto 3D, la asignación a cada marcador creado para determinado objeto 3D es una de las primeras formas de trabajo en realidad aumentada y una de las más factibles para esta aplicación.

El segundo sistema elegido es el sistema de Reconocimiento de formas, este sistema se eligió ya que de forma similar al de reconocimiento de marcadores busca una imagen para poder activar el la realidad aumentada, pero este sistema tiene mejoras ya que no solo se basa a marcadores creados sino que se puede capturar una imagen determinada que no sea marcador y vincularla a un objeto 3D y así poder mejorar la forma de activación de realidad aumentada de cada objeto de una manera más entretenida y menos abstracta.

De esta manera se eligió a los dos sistemas de Realidad Aumentada a utilizar, pero se recalca que se utilizó principalmente el sistema de Reconocimiento de marcadores, ya por circunstancias de tiempo dentro del desarrollo del proyecto no se pudo desarrollar el otro sistema pero cuenta con la raíz a la extensión y mejora con el sistema de reconocimiento de formas.



## Capítulo IV PLANIFICACIÓN

En éste capítulo se detalla la planificación del proyecto así como también los miembros participantes del proyecto basados en la metodología elegida para el desarrollo de la aplicación la cual es Scrum, del mismo modo se menciona la planificación de los requisitos, la lista de objetivos, luego se eligen los requisitos priorizados basados en el tiempo que toma realizar cada uno de ellos, y todos los pasos a seguir dentro de la iteración de Scrum.

**Roles Scrum:** En el Proyecto vamos a tener los siguientes participantes.

- Product owner: Amado Laban Peña
- Scrum manager: Eysson Francisco Saucedo García
- Scrum team: Eysson Francisco Saucedo García

### **Iteraciones:**

Release 1: Se ha defino un solo Sprint para el siguiente Proyecto ya que no hay muchos Objetivos u historias de usuario.

Sprint 1:      Fecha de inicio:      23/03/2015

Fecha de fin:      30/04/2015



### **Esfuerzo:**

Es el valor en horas que el Scrum Team realizará por día.

Se decidió trabajar 8 horas diarias.

Scrum Team:        8 horas x 30 días= 240 horas

En este capítulo se recogerán y analizarán todos los requisitos y requerimientos por parte de la empresa Importaciones Labán S.A.C- Huancabamba que serán necesarios para el desarrollo de la aplicación.

## **IV.1 PLANIFICACIÓN DE LOS REQUISITOS**

El primer día de la iteración se realizó la reunión de planificación de la iteración. La cual tiene dos partes:

Selección de requisitos (4 horas). El cliente presentó al equipo la lista de requisitos priorizada del producto o proyecto. El equipo preguntó al cliente las dudas que surgieron y seleccionó los requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados en el tiempo planificado.

Planificación de la iteración (4 horas máximo). El equipo elaboró la lista de tareas de la iteración necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo se hizo de manera conjunta y los miembros del equipo se auto asignaron las tareas.



## IV.2 LISTA DE OBJETIVOS

La lista de Objetivos o Historias de usuario son una lista priorizada de requerimientos funcionales, mejoras, tecnología y corrección de errores que deben incorporarse al producto a través de las sucesivas iteraciones llamadas Sprints, en el desarrollo de esta aplicación solo he decidido realizarla en una sola iteración o sprint ya que el proyecto no tiene mucha complejidad que requiera extender a más iteraciones por el momento, esta lista de Objetivos es creada con el cliente Importaciones Laban S.A.C ya que él es quien tiene que ver todas sus necesidades y requerimientos.

- Buscar la tecnología adecuada para dispositivos móviles(tablets y smartphone's)
- Elección de herramientas para el desarrollo de la aplicación (Software)
- Creación del Frontend
- Creación y búsqueda de objetos 3D
- Creación de marcadores
- Creación del Backend
- Colocación física de los marcadores.
- Pruebas con marcadores impresos

## IV.3 REQUISITOS PRIORIZADOS

En este punto a todos los requisitos o historias se le dio un valor con respecto al tiempo que se tardó en ejecutarse cada uno de estos para que luego poder desarrollarlos tomando en cuenta su valor en horas.



Nº	Historia de Usuario o requisito	Descripción	Tags	Valor en Horas
1	Buscar la tecnología adecuada para dispositivos móviles(tablets y smartphone's)	Se buscará el hardware adecuado que cumpla con los requisitos mínimos para que se pueda ejecutar la aplicación.		16:00
2	Elección de herramientas para el desarrollo de la aplicación (Software)	Se analizarán y elegirán las herramientas de software necesarias para poder realizar la aplicación de la manera más óptima posible.		24:00
3	Creación del FrontEnd	Se creará toda la parte visual de la aplicación como la imagen de inicio de la aplicación, el menú principal, etc.		72:00
4	Creación de Marcadores	Se buscarán o se crearán los marcadores que utilizaremos en la aplicación para proyectar en cada uno de ellos una determinada imagen.		36:00
5	Creación y búsqueda de objetos 3D	Se buscarán y crearán objetos 3D que se utilizarán para proyectarlos en la aplicación.		24:00
6	Creación del Backend	Se creará toda la parte funcional interna de la aplicación y dentro de esto tendremos los siguientes Tags:	#Agregar los objetos 3D, #Funcionalidad de la Cámara, #Detector de Geolocalización, #Escaneo y reconocimiento de marcadores	80:00
7	Colocación física de los marcadores	Se hará la impresión en físico de los marcadores y se colocarán para luego utilizarlos en las pruebas.		2:00
8	Prueba con marcadores impresos	Prueba de Proyección de Imágenes 3D asociada a cada marcador		4:00





**Tabla 1. Requisitos Priorizados Fuente: Elaboración Propia**

#### **IV.4 EJECUCIÓN DE ITERACIÓN**

Cada día el equipo realiza una reunión de sincronización (15 minutos máximos). Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:

- ✓ ¿Qué he hecho desde la última reunión de sincronización?
- ✓ ¿Qué voy a hacer a partir de este momento?
- ✓ ¿Qué impedimentos tengo o voy a tener?

Durante la iteración el Facilitador (Scrum Master) se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.

Elimina los obstáculos que el equipo no puede resolver por sí mismo.

Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.

#### **IV.5 INSPECCIÓN Y ADAPTACIÓN**

El último día de la iteración se realiza la reunión de revisión de la iteración donde se reunirá todo el trabajo realizado para ser inspeccionado y adaptado entre sí cada parte finalizada del trabajo.



## **IV.6 DEMOSTRACIÓN DE REQUISITOS**

Una vez que se haya terminado con la implementación y ejecución de todos los requisitos o historias de usuario, estos se presentarán al cliente para que vea los resultados del producto una vez finalizado el Sprint. Esta demostración de requisitos está sujeta a posibles modificaciones por el cliente ya que puede encontrar algunas deficiencias o querer algo nuevo en la aplicación lo que conllevaría a realizar un nuevo Sprint o iteración esta reunión tiene una duración de 4 horas.

### **IV.6.1 Retrospectiva**

El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de ir eliminando los obstáculos identificados. Esta reunión tiene una duración de 4 horas máximo.



## **Capítulo V IMPLEMENTACIÓN Y EVALUACIÓN DE RESULTADOS**

En este capítulo se menciona el desarrollo y la evaluación de los resultados, con respecto a la elaboración de la aplicación se describen los pasos a seguir desde el frontend hasta el backend, las herramientas a utilizar y se describen las características y funcionalidades de algunas de estas herramientas; también se menciona la constatación de los resultados obtenidos con la aplicación en la empresa Importaciones Laban S.A.C. – Huancabamba.

### **V.1 IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA**

También en este capítulo se documenta el desarrollo de la aplicación y también la prueba del funcionamiento de la aplicación construida, para esto se desarrollará de acuerdo a los requisitos o historias de usuario ya mencionadas en el capítulo anterior.

#### **V.1.1 Buscar la tecnología adecuada para dispositivos móviles (tablets y smartphone's)**

Se realizó la búsqueda de varios equipos Celulares (Smartphone's) y tabletas con sistema operativo Android y con algunas características mínimas de cámara y procesador, para que se ejecutar a aplicación que se creará.



## Celular SmartPhone:

Sistema Operativo	Android OS,v4.0.4 Ice Cream Sandwich
Cámara	8 MP, 3264x2448 pixeles
Memoria	16GB Memoria Interna, 1GB RAM
Procesador	1.4 GHz, GPU Mali 400MP

**Tabla 2. Características de Smartphone Samsung Galaxy**

**Fuente:** <http://www.smart-gsm.com/moviles/samsung-galaxy-s3> (Samsung, 2014)

## Tablet (Tableta):

Sistema Operativo	Android OS, v4.2 Jelly Bean
Cámara	2 MP, WSVGA (1.024 x 600, 169 PPI)
Memoria	RAM de 1 GB, almacenamiento de 8 GB
Procesador	1,2 GHz,

**Tabla 3. Características de Smartphone Samsung Galaxy**

**Fuente:** <http://www.samsung.com/latin/consumer/mobile-devices/tablets/galaxy-tab/SM-T110NDWATPA>



## V.1.2 Elección de herramientas para el desarrollo de la aplicación (Software)

### Herramientas Scrum:

Kunagi. Herramienta Web que proporciona un sistema de gestión de proyectos basado en Scrum. Ofrece herramientas colaborativas y otras facilidades, como un cuadro de mando del proyecto, un panel interactivo para el Sprint o soporte a la estimación con Planning Poker.



**Figura 9. Logo Kunagi**

**Fuente:** (kunagi, 2015)

ScrumDo. Herramienta Scrum muy centrada en la simplicidad y en la facilidad de uso. Permite gestionar las listas de tareas e historias de usuario, crear y gestionar iteraciones, obtener gráficos de avance “burndown” y también dar soporte a la estimación con Planning Poker.



**Figura 10. Logo ScrumDo**

**Fuente:** (*Scrumdo, 2015*)

SprintoMeter. Herramienta para la gestión, medición y seguimiento de proyectos Scrum y eXtreme Programming. Para simplificar el intercambio de datos permite exportar gráficos e informes a Excel. Posee gráficos de avance burndown en 3D.

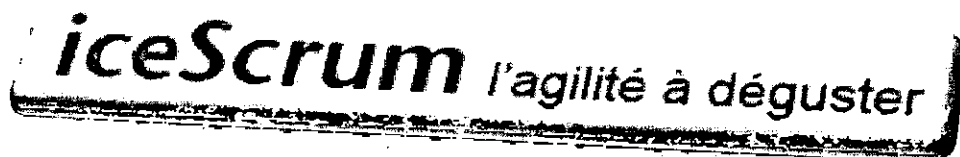


**Figura 11. Logo Sprintometer**

**Fuente:** (*Sprintometer, 2015*)



IceScrum. Herramienta Scrum y Kanban. Ofrece las opciones de operación, consulta y estimación de historias de usuario. Permite añadir historias de usuario a la pila de producto, dividir el tiempo en Sprints y mover estas historias de la pila de producto a cada uno de los Sprint. Posee la técnica de Planning Poker para la estimación y paneles virtuales.



**Figura 12.** Logo iceScrum

**Fuente:** (Icescrum, 2015)



Pango Scrum. Otra de las herramientas Scrum online, con una interfaz sencilla y amigable que permite escribir, estimar y priorizar la pila de producto. Facilita en gran medida la planificación de Sprints y las reuniones.



**Figura 13. Logo Pango Scrum**

**Fuente:** (*PangoScrum, 2015*)

➤ Luego de revisar las herramientas anteriores, se eligió a ScrumDo por su fácil manejo y entendimiento además de poseer mucha flexibilidad en su uso en cada una de sus tareas (Historias de usuario) y mediciones de avance con sus reportes gráficos y es de uso gratuito.

➤ Primero se procedió a crear un proyecto en ScrumDo para poder llevar el control del proyecto en Scrum, a continuación se va a explicar el funcionamiento de algunas herramientas que usaremos de esta aplicación.





Figura 14. ScrumDo

Fuente ScrumDo.com

### Backlog:

En esta opción del ScrumDo se va a agregar las historias de usuario o requisitos que ya previamente les hemos asignado los sus valores respectivos con respecto al tiempo que tarda en hacerse cada uno de estos.

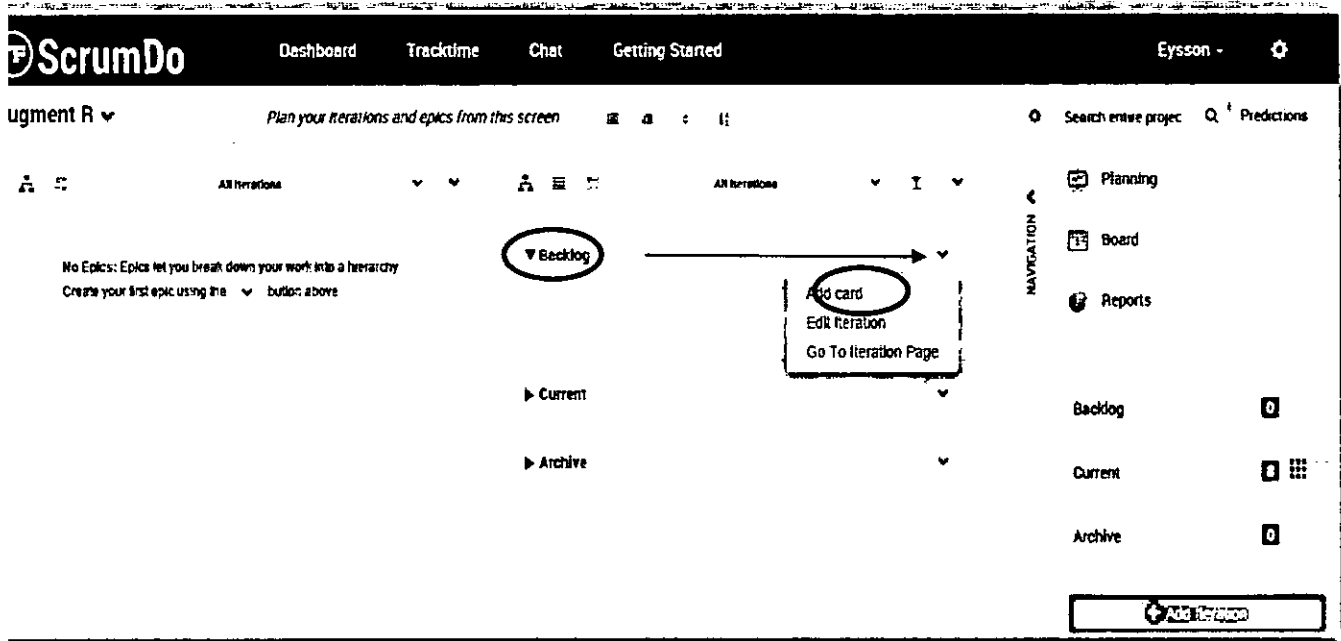


Figura 15. Backlog

Fuente: Creación Propia



Para esto se va al combo box que está a la derecha de Backlog y se selecciona Add Card con lo cual se abrirá la siguiente ventana para agregar la lista de los requisitos o historias de usuario.

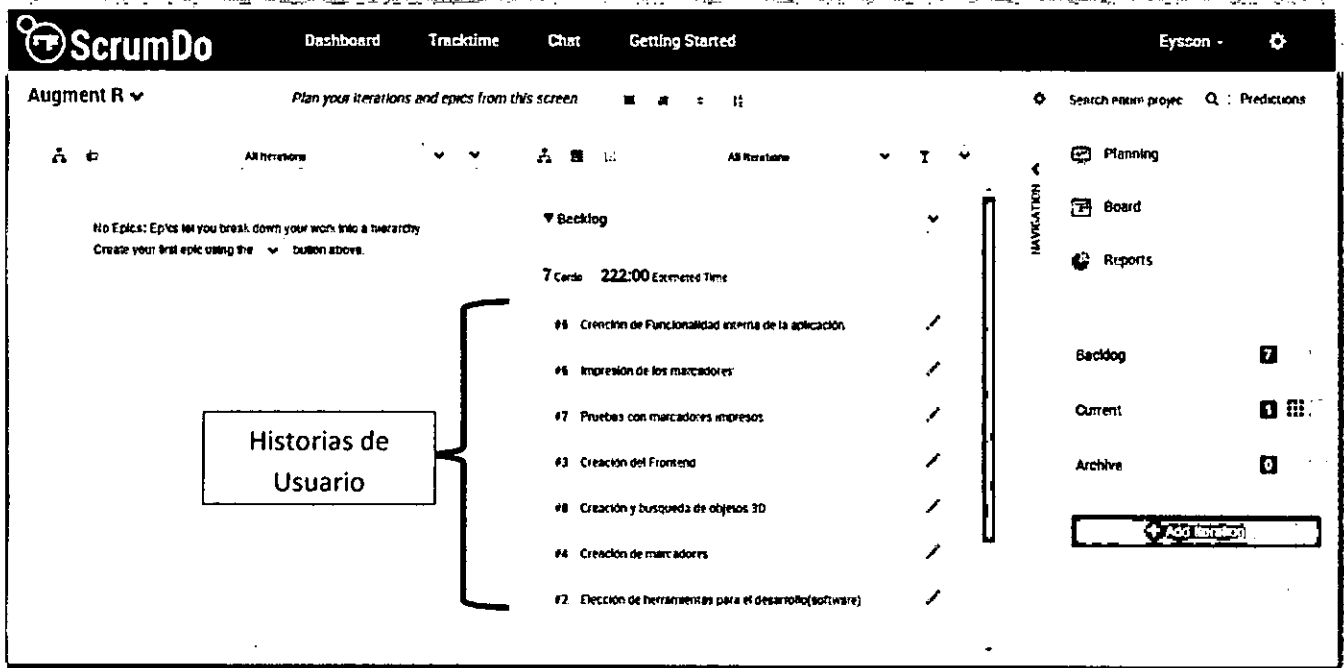


Figura 16. Backlog 1 Historias de Usuario

Fuente: Elaboración propia

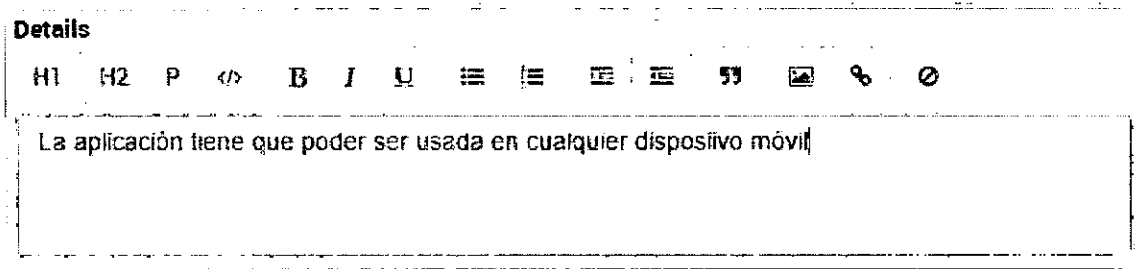


The screenshot shows a Jira backlog item card. At the top, there are buttons for 'Planning Poker', 'Track Time', 'Duplicate', 'Convert to Epic', 'Delete', and 'Tasks', along with 'Save' and 'Cancel' buttons. The card title is '#5 Creación de Funcionalidad interna de la aplicación', with a callout box labeled 'Título' pointing to it. Below the title, there are sections for 'Details', 'Labels (0)', '#Tags (5)', and 'Assignees'. The 'Details' section includes 'Backlog', '2 Points', 'No Epic', and '80:00'. The 'Tags' section lists '#Agregar los Objetos 3D de la Cámara', '#Funcionalidad de geolocalización', '#Escaneo marcadores', and '#Reconocimiento de marcadores'. The 'Assignees' section shows 'None Selected'. The 'Comments' section has a 'Write a comment' button. The 'History' section shows a comment by 'Eysson Saucedo' dated 'FEB 25, 2018 4:40 PM'.

**Figura 17. Backlog 2 From de creación de Historia de Usuario**

**Fuente: Creación propia**

Acá se encuentra al lado izquierdo el número de la historia que se va agregando y a la derecha se tiene una caja de texto para agregar el título del requisito.

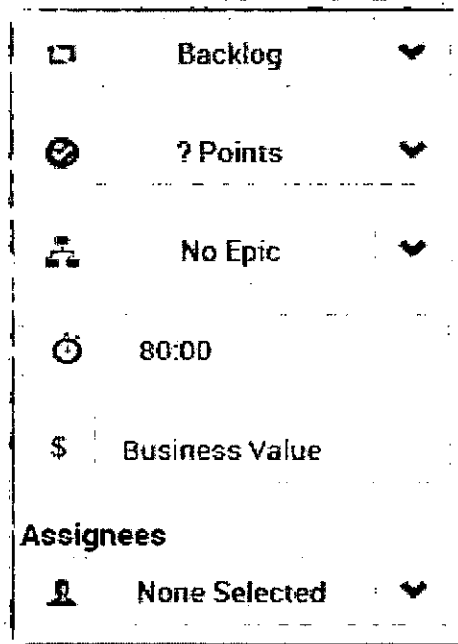


**Figura 18. Detalle de Historia**

**Fuente: Elaboración Propia**

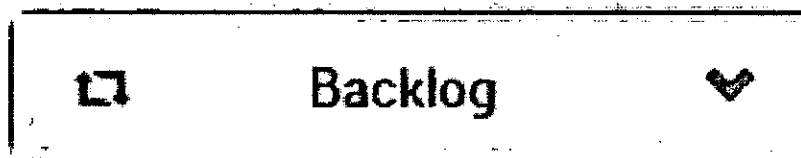
De bajo del título se encuentra Detalles (Details) acá se agrega los detalles de cada Requerimiento o historia de usuario que se van a ir agregando.

A la parte derecha se encuentra un menú en donde hay varias opciones entre estas están actualizar estado, Points, Hora y también estará Assignees.



**Figura 19. Menú de Rótulo**

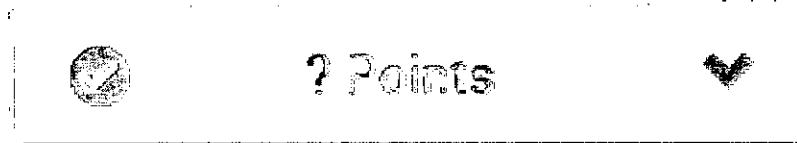
**Fuente: Elaboración Propia**



**Figura 20. Estado Historia**

**Fuente: Scrumco.com**

**Estado:** Este combo box contiene el estado de donde se puede encontrar esta historia de usuario, puede estar en el Backlog en Proceso (Current) o Archivado, dependiendo del avance de la historia de usuario.



**Figura 21. Puntos de Historia**

**Fuente: scrumdo.com**

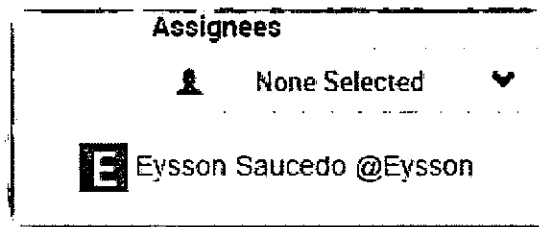
**Points:** En esta opción se encuentra el valor en puntos que también se le puede asignar a cada historia de usuario para priorizarlas a la hora de desarrollarlas, pero en este caso no se ha utilizado.



**Figura 22. Tiempo de Historia**

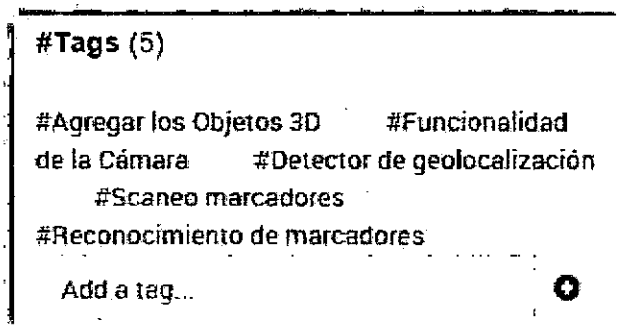
**Fuente: Elaboración Propia**

**Time:** Esta opción permite ingresar el tiempo en horas que se asignen a cada una de las historias de usuario que se vayan agregando.



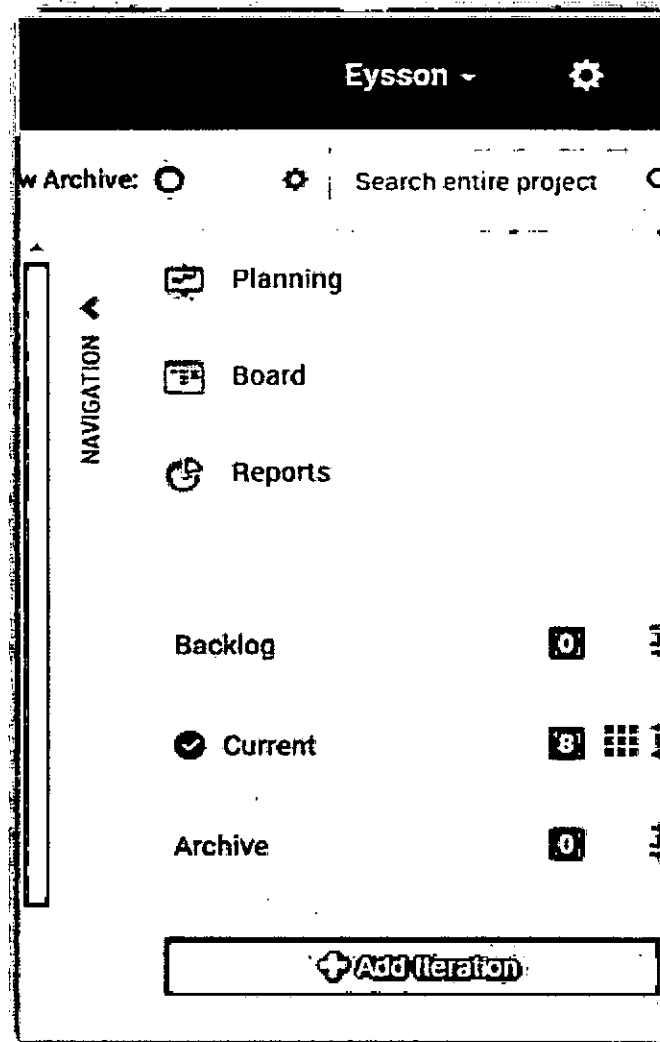
**Figura 23. Assignees Team**  
**Fuente: Elaboración Propia**

**Assignees:** Acá se elegirá a quien de los miembros del equipo se encargara del desarrollo total de dicha historia de usuario.



**Figura 24. Tags de Historia de Usuario**  
**Fuente: Elaboración propia**

**Tags:** En esta opción se puede ingresar los puntos clave que se desarrollaran dentro de la historia de usuario.

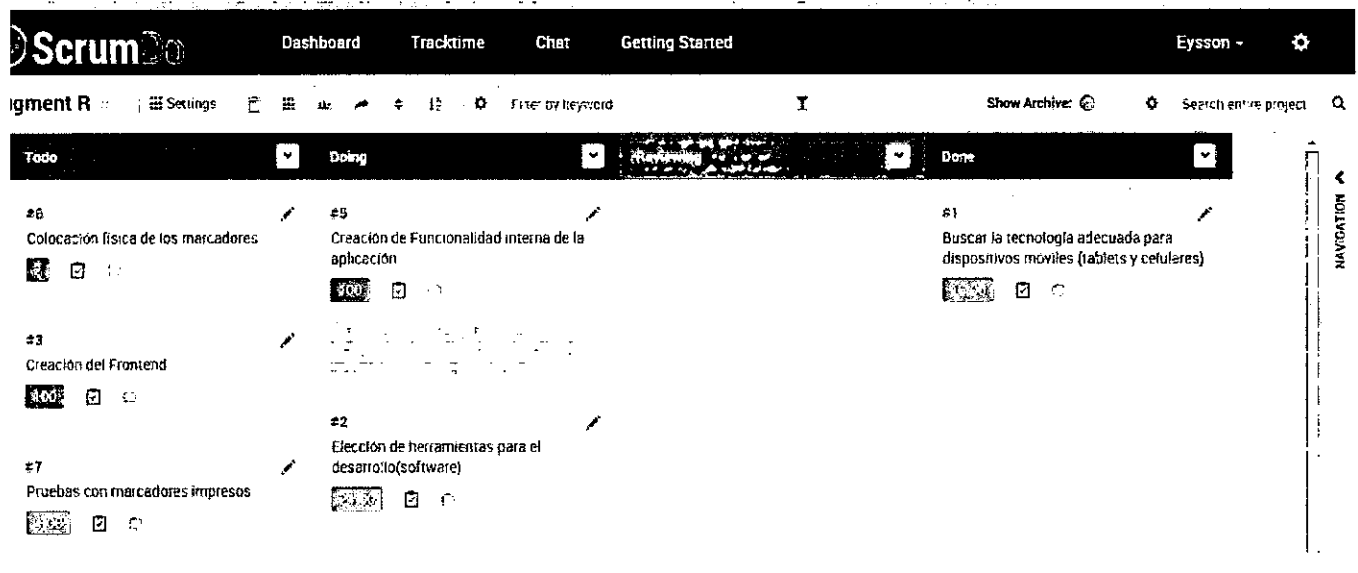


**Figura 25. Navigation 1 Menú de Navigation (Navegación)**

**Fuente Elaboración Propia**

**Navigation:** A la parte derecha del proyecto se encuentra el menú desplegable de nombre Navigation (Navegación) en el cual se encuentra Planning (Planificación), Board (Pizarra), Reports (Reportes) y también se puede ver el número de historias que están en el Backlog, las que están en proceso (Current) y las que están archivadas.





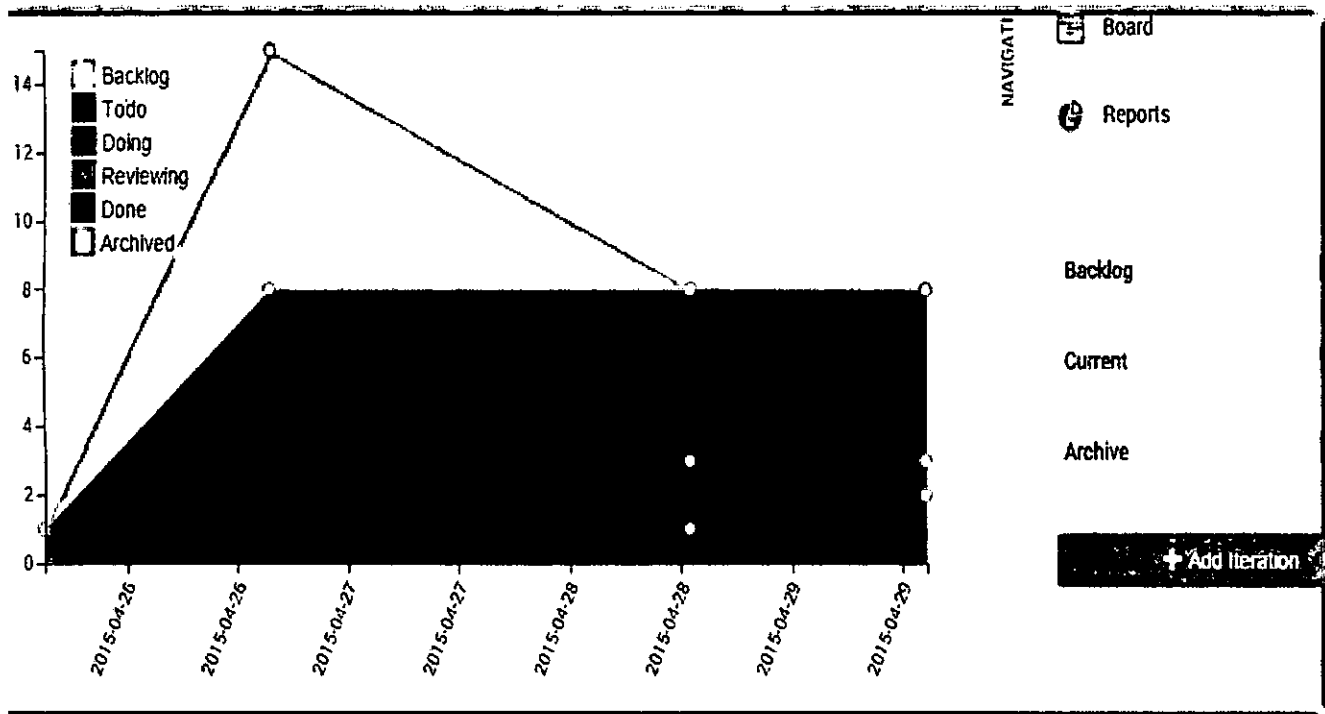
**Figura 26. Board 1 Pizarra de Trabajo**

**Fuente: Elaboración Propia**

En esta Pizarra (Board), se colocan todas las historias de usuario que intervienen en el Sprint y luego se van ubicando cada una de ellas en donde se encuentren actualmente, esto se realiza siguiendo una secuencia al inicio la tarea se pondrá en desarrollo (Doing) una vez concluida pasa a revisión (Reviewing) y finalmente pasa a hecho (Done).



## Reports:



**Figura 27. Reportes de Avance del Sprint**

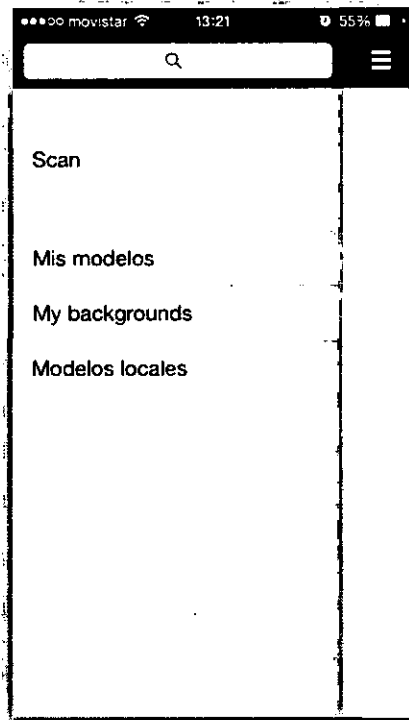
**Fuente: Elaboración Propia**

En esta opción se muestra un reporte que se expresa en un gráfico sobre el avance de las historias de usuario realizadas o Sprint del proyecto que se está ejecutando.



### V.1.3 Creación del FrontEnd

El Frontend tendrá el siguiente modelo en su menú principal, este es un Sliding menú con Navigation Drawer, un menú que se despliega del lado izquierdo de la pantalla mostrando el menú. Esta tecnología se eligió porque es una de las más recientes, más fácil de usar, intuitiva y atractiva al usuario.

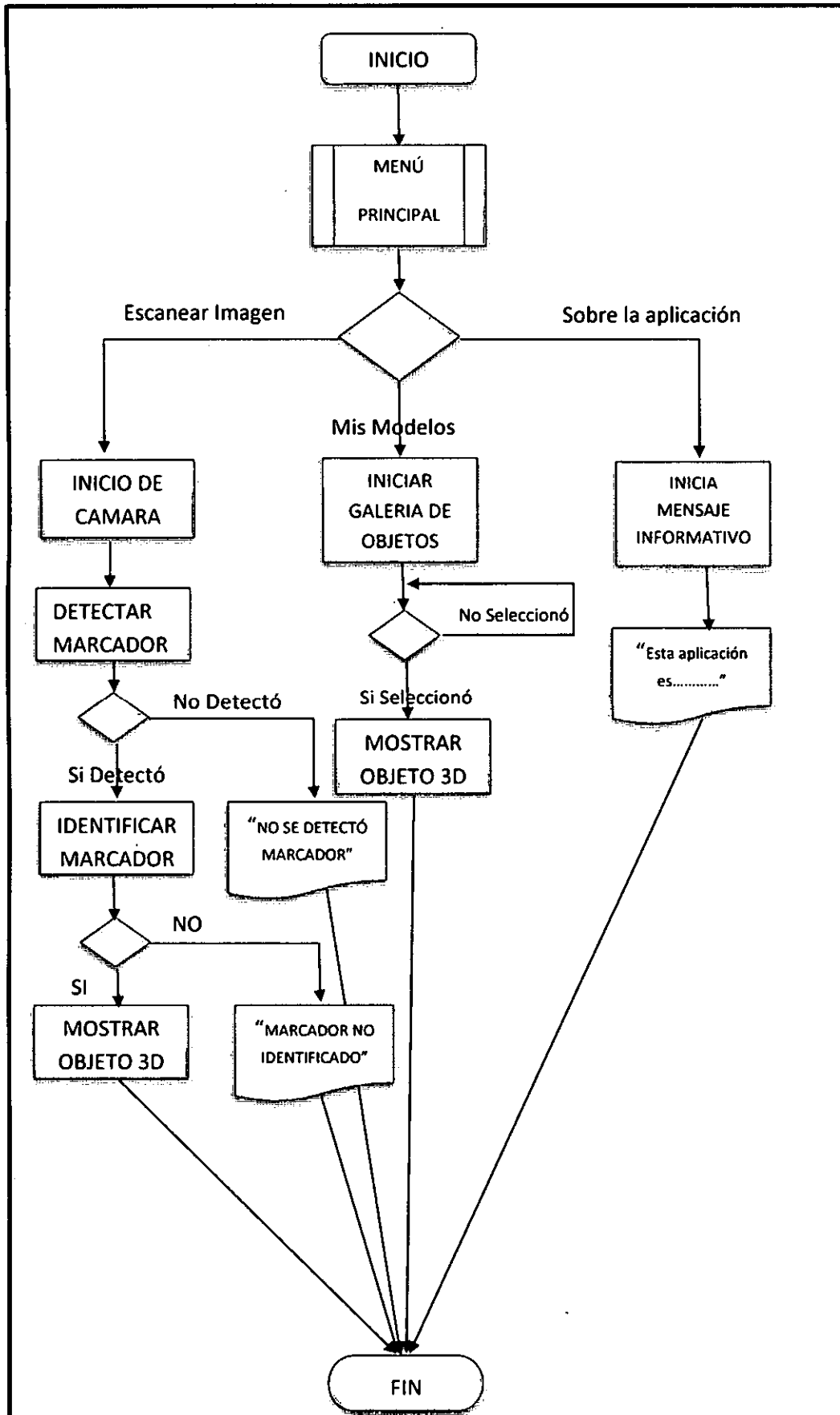


**Figura 28. Menú Principal**

**Fuente: Elaboración Propia**



## Diagrama de Flujo de la Aplicación





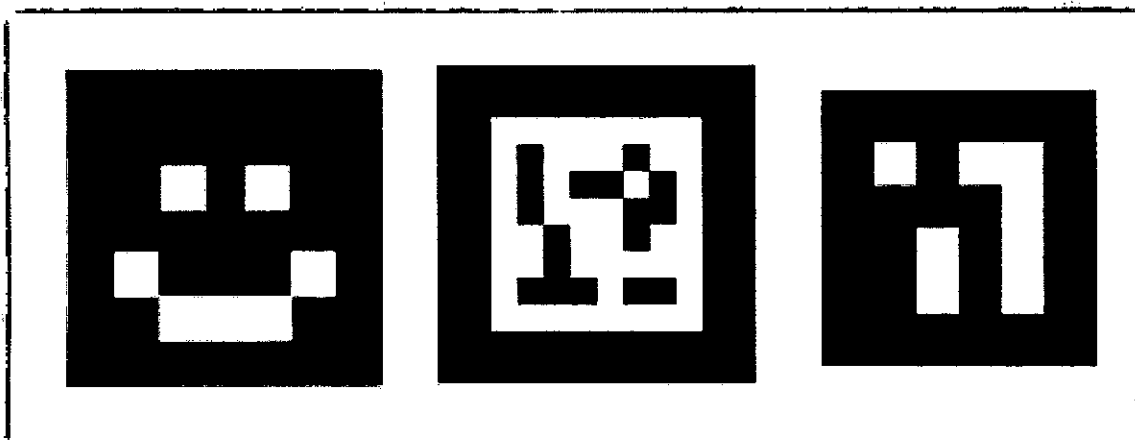
## V.1.4 Creación de Marcadores

Para la creación de los marcadores se utilizó la herramienta Unity con el cual se facilita la creación de los mismos, una vez que se hallan creado todos los marcadores que se usarán quedarán listos para ser asociados a un determinado objeto 3D que más adelante se obtuvieron.



Figura 29. Marcador1

Fuente: Elaboración Propia



**Figura 30. Marcadores**

**Fuente: Elaboración Propia**

Acá se muestra ya los marcadores creados con Unity que se asignaron a un determinado Objeto 3D, para poder visualizarlos en realidad aumentada.

### **V.1.5 Creación y búsqueda de objetos 3D**

Los objetos 3D se buscaron en internet ya que para diseñar desde cero cada uno de ellos demandaría de más tiempo en el desarrollo de la aplicación. Para la búsqueda se consultó previamente los modelos de muebles que disponía en la tienda y luego se procedió a la búsqueda en internet y se encontró los necesarios.

Estos Objetos 3D se encontraron con su diseño y sus respectivos detalles de cada uno de ellos, ya que muchas veces solo se puede encontrar modelos sin detalles como color o texturas, la búsqueda tomo un tiempo regular pero finalmente se encontraron.



Cama de madera de 2 plazas con fino acabado y cabecera tallada.

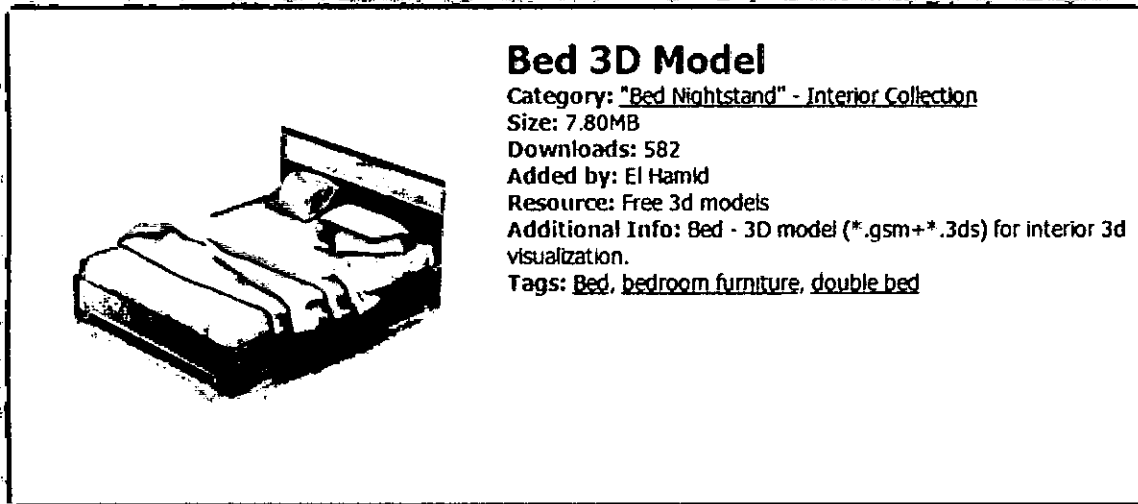


Figura 31. Objeto 3D Cama

Fuente: (3D, 2015)

Escritorio de madera para oficina con tres cajones y fino acabado.

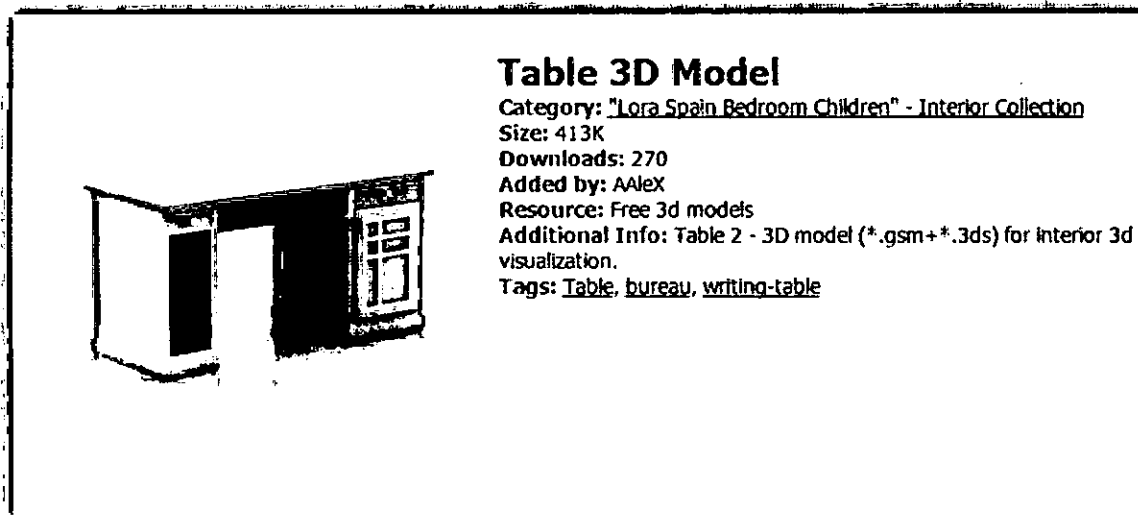
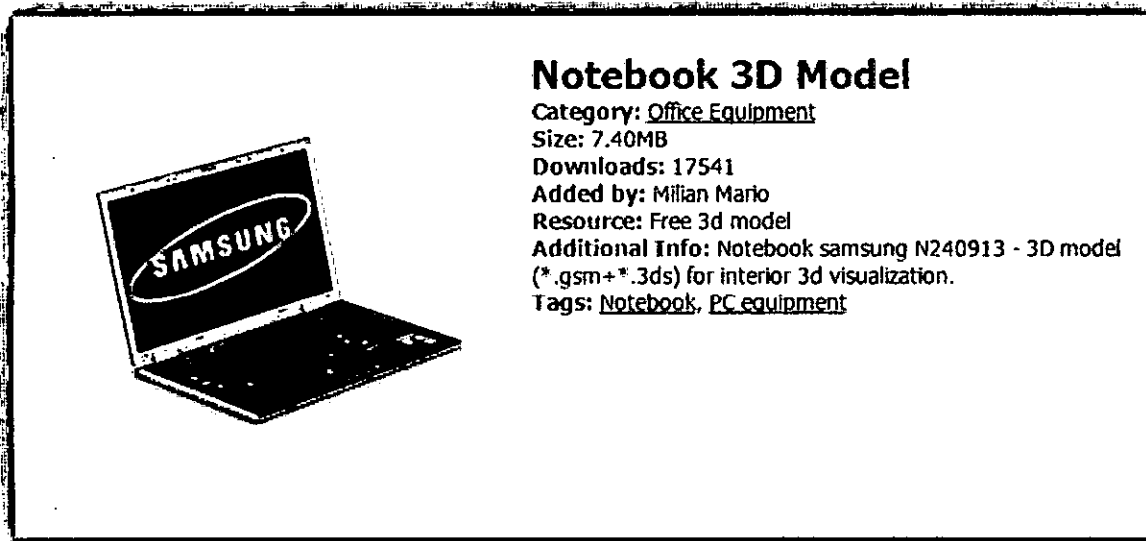


Figura 32. Objeto 3D Escritorio

Fuente: <http://archive3d.net/?a=download&id=7e0bd296>



## Laptop Samsung Core i3 de tercera generación con pantalla de 14"



**Figura 33. Objeto 3D Laptop Samsung**

**Fuente:** <http://archive3d.net/?a=download&id=0a1cd0f0>

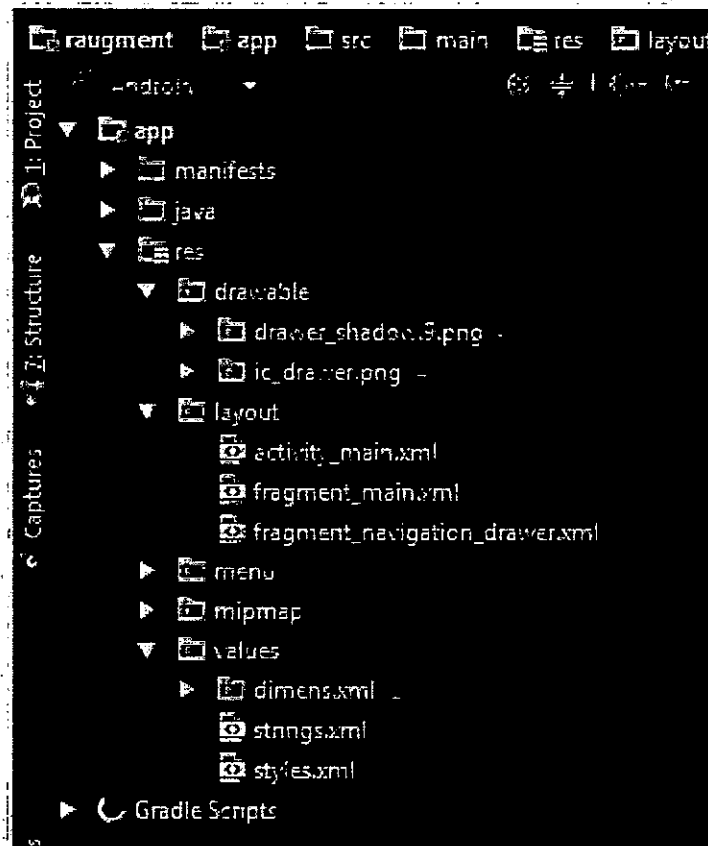
### V.1.6 Creación del Backend

A continuación la creación del backend; previamente ya se han creado las vistas del Frontend de la Aplicación, luego se inicia con la estructura del proyecto que se desarrolló en Android Studio.





Acá se muestra todos los componentes que forman parte del funcionamiento interno de la aplicación, para el desarrollo de la aplicación para el sistema operativo Android, es necesario tener un manifiesto donde se encuentran la actividades a ejecutarse, los permisos que requiere la aplicación, las versiones a las cuales está dirigida la aplicación, entre otras características del programa. Aquí se instancia que actividad será la que se ejecute al iniciar la aplicación, en este caso es Menu inicial y se nombra a las actividades que se ejecutaran cuando sean llamadas (MainActivity, Imagenes, Texto). Así mismo podemos observar que para que sea aplicable a un amplio número de dispositivos, con versiones tanto antiguas como nuevas, la aplicación soporta desde el nivel de API 16 (Versión 4.1 Jelly Bean) hasta API 22 (Versión 5.1 Lollipop).



**Figura 34. Estructura de la Aplicación**

**Fuente: Elaboración Propia**

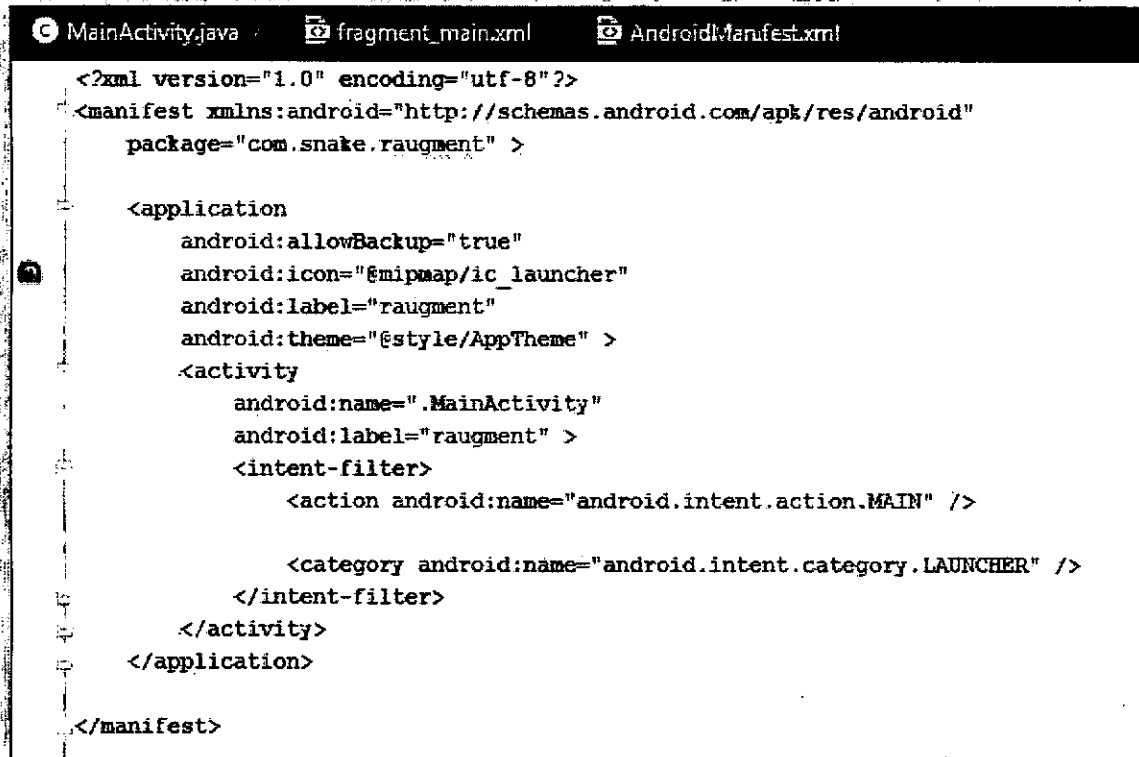


El archivo de manifiesto o su nombre Manifest en Inglés presenta información esencial acerca de su aplicación para el sistema Android, la información del sistema que debe tener antes de que pueda ejecutar cualquiera de código de la aplicación. Entre otras cosas, el manifiesto hace lo siguiente:

- ✓ Nombra el paquete Java para la aplicación. El nombre del paquete sirve como identificador único para la aplicación.
- ✓ En él se describen los componentes de la aplicación - las actividades, servicios, receptores de radiodifusión, y proveedores de contenido que la aplicación se compone de. Nombra a las clases que implementan cada uno de los componentes y publica sus capacidades (por ejemplo, que Intención mensajes que pueden manejar). Estas declaraciones permiten el sistema Android sabe lo que los componentes son y en qué condiciones pueden ser lanzados.
- ✓ Determina que proceso acogerá componentes de la aplicación.
- ✓ Declara que los permisos que la aplicación debe tener para acceder a partes protegidas de la API e interactuar con otras aplicaciones.
- ✓ También declara los permisos necesarios a otros a tener a fin de interactuar con los componentes de la aplicación.
- ✓ En él se enumeran las Instrumentación clases que proporcionan perfiles y otra información que se ejecuta la aplicación. Estas declaraciones están presentes en el manifiesto sólo mientras la aplicación está siendo desarrollada y probada; que están retirados antes de la publicación de la solicitud.



- ✓ Declara el nivel mínimo de la API de Android que la aplicación requiere.
- ✓ En él se enumeran las bibliotecas que la solicitud debe ser vinculado en contra.



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.snake.raugment" >

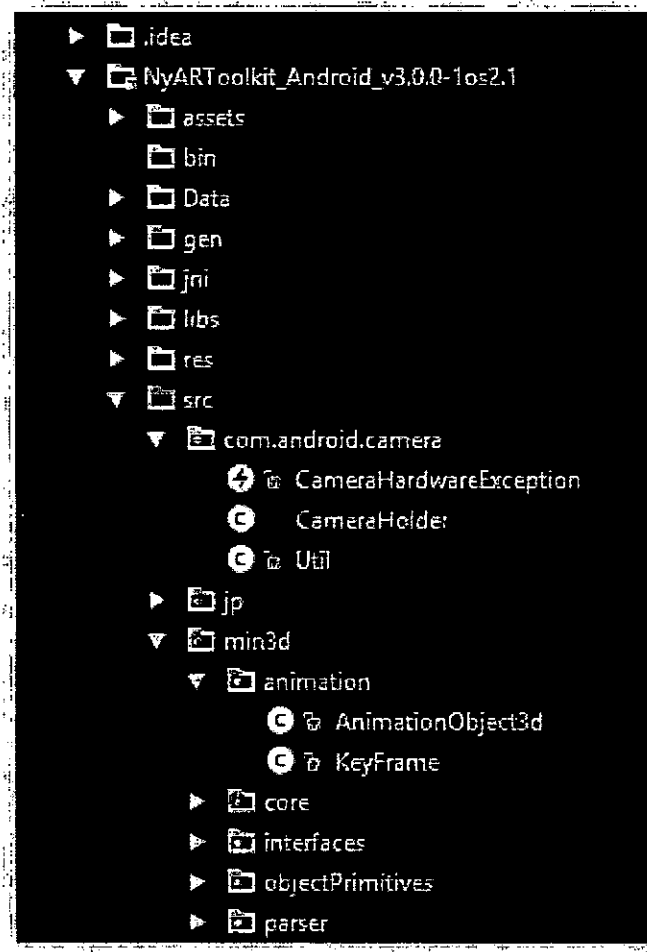
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="raugment"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="raugment" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

**Figura 35. Archivo AndroidManifest.xml**

**Fuente: Elaboración Propia**



**Figura 36. Estructura del proyecto con clases de cámara y animación 3d**

**Fuente: Elaboración Propia**

Dentro del Backend también se desarrolla la vinculación entre los marcadores y los objetos 3D, este proceso se desarrolla dentro del programa Unity el cual nos facilita hacer este trabajo para luego agregarlo al código en Android Studio.



**Clase Camera Holder:** Esta clase es la que se encarga del funcionamiento de la cámara para realidad aumentada en el proyecto de Android. Esta clase es la encargada de hacer funcionar la cámara del dispositivo para que se pueda trabajar con la realidad aumentada.

```
import static com.android.camera.Util.Assert;

import android.hardware.Camera.Parameters;
import android.os.Build;
import android.os.Handler;
import android.os.HandlerThread;
import android.os.Looper;
import android.os.Message;
import android.util.Log;

import java.io.IOException;

public class CameraHolder {
    private static final String TAG = "CameraHolder";
    private android.hardware.Camera mCameraDevice;
    private long mKeepBeforeTime = 0; // Keep the Camera before this
time.
    private final Handler mHandler;
    private int mUsers = 0; // number of open() - number of release()

    // We store the camera parameters when we actually open the
device,
    // so we can restore them in the subsequent open() requests by the
user.
    // This prevents the parameters set by the Camera activity used by
// the VideoCamera activity inadvertently.
    private Parameters mParameters;

    // Use a singleton.
    private static CameraHolder sHolder;
    public static synchronized CameraHolder instance() {
        if (sHolder == null) {
            sHolder = new CameraHolder();
        }
        return sHolder;
    }

    private static final int RELEASE_CAMERA = 1;
    private class MyHandler extends Handler {
        MyHandler(Looper looper) {
            super(looper);
        }

        @Override
        public void handleMessage(Message msg) {
            switch(msg.what) {
                case RELEASE_CAMERA:
                    synchronized (CameraHolder.this) {
                        // In 'CameraHolder.open', the
```



```
'RELEASE CAMERA' message
// will be removed if it is found in the
queue. However,
// there is a chance that this message has
been handled
// before being removed. So, we need to add a
check
// here:
if (CameraHolder.this.mUsers == 0)
releaseCamera();
    }
    break;
}
}
}

private CameraHolder() {
    HandlerThread ht = new HandlerThread("CameraHolder");
    ht.start();
    mHandler = new MyHandler(ht.getLooper());
}

public synchronized android.hardware.Camera open()
    throws CameraHardwareException {
    Assert(mUsers == 0);
    if (mCameraDevice == null) {
        try {
            mCameraDevice = android.hardware.Camera.open();
        } catch (RuntimeException e) {
            Log.e(TAG, "fail to connect Camera", e);
            throw new CameraHardwareException(e);
        }
        mParameters = mCameraDevice.getParameters();
//    } else {
//        try {
//            mCameraDevice.reconnect();
//        } catch (IOException e) {
//            Log.e(TAG, "reconnect failed.");
//            throw new CameraHardwareException(e);
//        }
//        mCameraDevice.setParameters(mParameters);
    }
    ++mUsers;
    mHandler.removeMessages(RELEASE_CAMERA);
    mKeepBeforeTime = 0;
    return mCameraDevice;
}

/**
 * Tries to open the hardware camera. If the camera is being used
or
 * unavailable then return {@code null}.
 */
public synchronized android.hardware.Camera tryOpen() {
    try {
        return mUsers == 0 ? open() : null;
    } catch (CameraHardwareException e) {
        // In eng build, we throw the exception so that test tool
        // can detect it and report it
        if ("eng".equals(Build.TYPE)) {
            throw new RuntimeException(e);
        }
    }
}
```



```
    }
    return null;
}

public synchronized void release() {
    Assert(mUsers == 1);
    --mUsers;
    mCameraDevice.stopPreview();
    releaseCamera();
}

private synchronized void releaseCamera() {
    Assert(mUsers == 0);
    Assert(mCameraDevice != null);
    long now = System.currentTimeMillis();
    if (now < mKeepBeforeTime) {
        mHandler.sendMessageDelayed(RELEASE_CAMERA,
            mKeepBeforeTime - now);
        return;
    }
    mCameraDevice.release();
    mCameraDevice = null;
}

public synchronized void keep() {
    // We allow (mUsers == 0) for the convenience of the calling
    activity.
    // The activity may not have a chance to call open() before
    the user
    // choose the menu item to switch to another activity.
    Assert(mUsers == 1 || mUsers == 0);
    // Keep the camera instance for 3 seconds.
    mKeepBeforeTime = System.currentTimeMillis() + 3000;
}
}
```



**Clase AnimationObject3d:** En esta clase se encuentra todo lo relacionado con la animación 3D, dentro de esta clase se trabaja con los parámetros y ubicación de la realidad captados por la cámara para poder proyectar la animación del Objeto 3D en un entorno ya establecido por la clase.

```
package min3d.animation;

import min3d.core.FacesBufferedList;
import min3d.core.Object3d;
import min3d.core.TextureList;
import min3d.core.Vertices;

public class AnimationObject3d extends Object3d {
    private int numFrames;
    private KeyFrame[] frames;
    private int currentFrameIndex;
    private long startTime;
    private long currentTime;
    private boolean isPlaying;
    private float interpolation;
    private float fps = 70;
    private boolean updateVertices = true;
    private String currentFrameName;
    private int loopStartIndex;
    private boolean loop = false;

    public AnimationObject3d(int $maxVertices, int $maxFaces, int
$numFrames) {
        super($maxVertices, $maxFaces);
        this.numFrames = $numFrames;
        this.frames = new KeyFrame[numFrames];
        this.currentFrameIndex = 0;
        this.isPlaying = false;
        this.interpolation = 0;
        this._animationEnabled = true;
    }

    public AnimationObject3d(Vertices $vertices, FacesBufferedList
$faces, TextureList $textures, KeyFrame[] $frames)
    {
        super($vertices, $faces, $textures);
        numFrames = $frames.length;
        frames = $frames;
    }

    public int getCurrentFrame() {
        return currentFrameIndex;
    }

    public void addFrame(KeyFrame frame) {
        frames[currentFrameIndex++] = frame;
    }

    public void setFrames(KeyFrame[] frames) {
        this.frames = frames;
    }
}
```





```
}

public void play() {
    startTime = System.currentTimeMillis();
    isPlaying = true;
    currentFrameName = null;
    loop = false;
}

public void play(String name) {
    currentFrameIndex = 0;
    currentFrameName = name;

    for (int i = 0; i < numFrames; i++) {
        if (frames[i].getName().equals(name)) {
            loopStartIndex = currentFrameIndex = i;
            break;
        }
    }

    startTime = System.currentTimeMillis();
    isPlaying = true;
}

public void play(String name, boolean loop) {
    this.loop = loop;
    play(name);
}

public void stop() {
    isPlaying = false;
    currentFrameIndex = 0;
}

public void pause() {
    isPlaying = false;
}

public void update() {
    if (!isPlaying || !updateVertices)
        return;
    currentTime = System.currentTimeMillis();
    KeyFrame currentFrame = frames[currentFrameIndex];
    KeyFrame nextFrame = frames[(currentFrameIndex + 1) %
numFrames];

    if (currentFrameName != null &&
!currentFrameName.equals(currentFrame.getName()))
    {
        if (!loop)
            stop();
        else
            currentFrameIndex = loopStartIndex;
        return;
    }

    float[] currentVerts = currentFrame.getVertices();
    float[] nextVerts = nextFrame.getVertices();
    float[] currentNormals = currentFrame.getNormals();
    float[] nextNormals = nextFrame.getNormals();
}
```



```
int numVerts = currentVerts.length;

float[] interPolatedVerts = new float[numVerts];
float[] interPolatedNormals = new float[numVerts];

for (int i = 0; i < numVerts; i += 3) {
    interPolatedVerts[i] = currentVerts[i] + interpolation *
(nextVerts[i] - currentVerts[i]);
    interPolatedVerts[i + 1] = currentVerts[i + 1] +
interpolation * (nextVerts[i + 1] - currentVerts[i + 1]);
    interPolatedVerts[i + 2] = currentVerts[i + 2] +
interpolation * (nextVerts[i + 2] - currentVerts[i + 2]);
    interPolatedNormals[i] = currentNormals[i] + interpolation *
(nextNormals[i] - currentNormals[i]);
    interPolatedNormals[i + 1] = currentNormals[i + 1] +
interpolation * (nextNormals[i + 1] - currentNormals[i + 1]);
    interPolatedNormals[i + 2] = currentNormals[i + 2] +
interpolation * (nextNormals[i + 2] - currentNormals[i + 2]);
}

interpolation += fps * (currentTime - startTime) / 1000;

vertices().overwriteNormals(interPolatedNormals);
vertices().overwriteVerts(interPolatedVerts);

if (interpolation > 1) {
    interpolation = 0;
    currentFrameIndex++;

    if (currentFrameIndex >= numFrames)
        currentFrameIndex = 0;
}

startTime = System.currentTimeMillis();
}

public float getFps() {
    return fps;
}

public void setFps(float fps) {
    this.fps = fps;
}

public Object3d clone(boolean cloneData)
{
    Vertices v = cloneData ? _vertices.clone() : _vertices;
    FacesBufferedList f = cloneData ? _faces.clone() : _faces;
    //KeyFrame[] fr = cloneData ? getClonedFrames() : frames;

    AnimationObject3d clone = new AnimationObject3d(v, f, _textures,
frames);
    clone.position().x = position().x;
    clone.position().y = position().y;
    clone.position().z = position().z;
    clone.rotation().x = rotation().x;
    clone.rotation().y = rotation().y;
    clone.rotation().z = rotation().z;
    clone.scale().x = scale().x;
    clone.scale().y = scale().y;
    clone.scale().z = scale().z;
}
```



```
clone.setFps(fps);
clone.animationEnabled(animationEnabled());
return clone;
}

public KeyFrame[] getClonedFrames()
{
    int len = frames.length;
    KeyFrame[] cl = new KeyFrame[len];

    for(int i=0; i<len; i++)
    {
        cl[i] = frames[i].clone();
    }

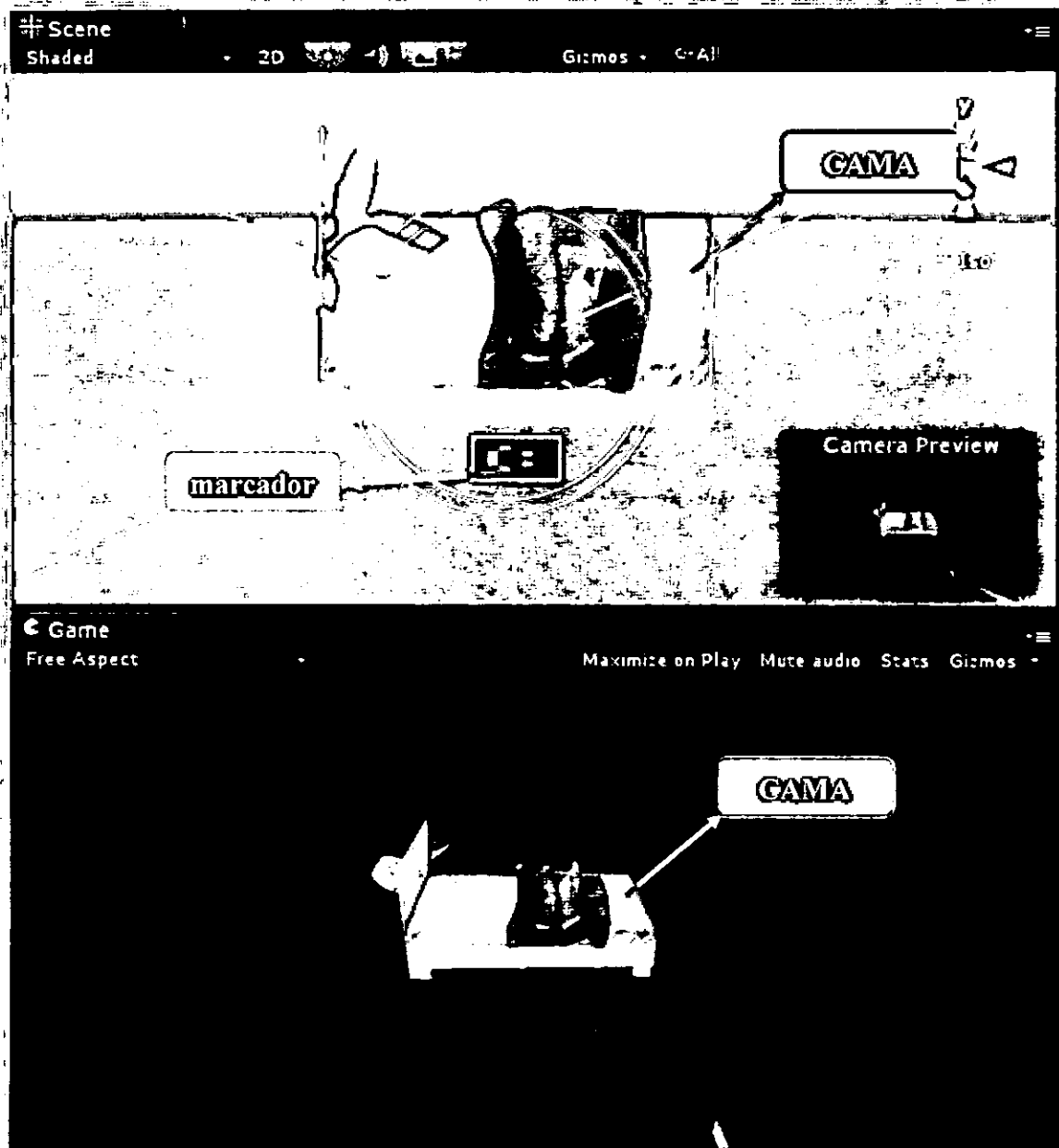
    return cl;
}

public boolean getUpdateVertices() {
    return updateVertices;
}

public void setUpdateVertices(boolean updateVertices) {
    this.updateVertices = updateVertices;
}
}
```



Ahora se muestra una imagen del proceso de vinculación en Unity.



**Figura 37. Asignando Objeto 3D al Marcador**

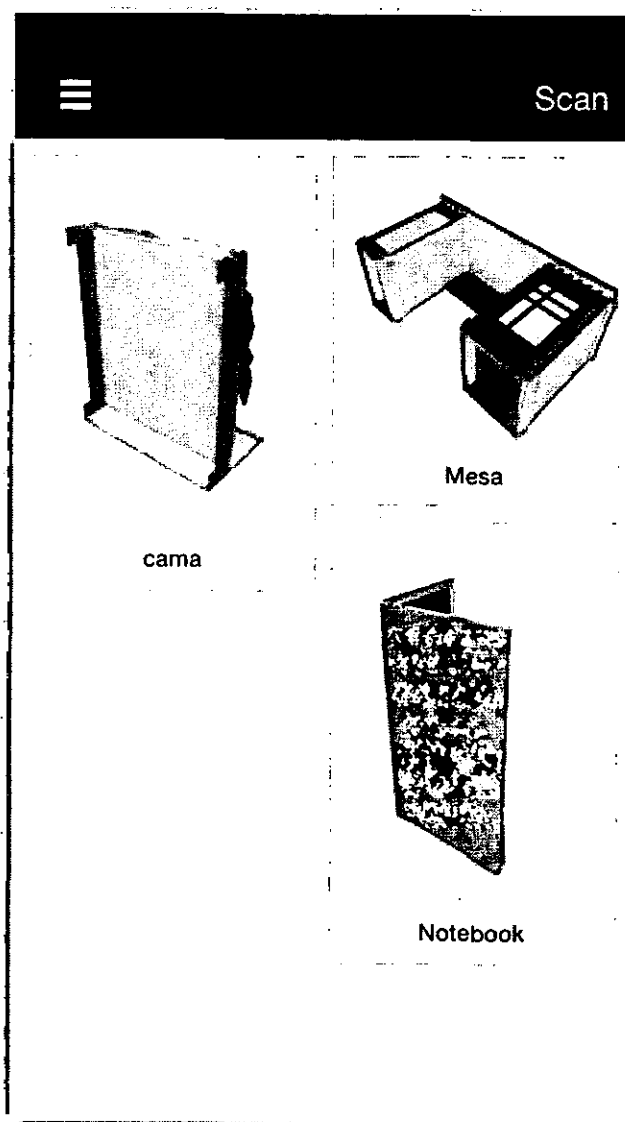
**Fuente: Elaboración Propia**

Una vez que ya se tienen los marcadores creados y los objetos 3D que se obtuvieron desde internet ahora la asignación o vinculación se realizó con Unity, luego este genera un archivo el cual lo agregaremos al proyecto en Android asignándole los permisos necesarios en el archivo Manifest.



Para mayor detalle sobre la vinculación del marcador con el objeto 3D se adjunta un manual de realidad aumentada con Unity en el Anexo 1, el cual facilitará el entendimiento y realización del mismo.

Una vez que ya se han agregado los marcadores vinculados a cada objeto 3D se agregan a la lista quedando de esta manera.



**Figura 38. Lista de Objetos 3D**

**Fuente: Elaboración Propia**



## INSTALACIÓN DE LA APLICACIÓN

Una vez que ya se ha creado la aplicación, exportamos el producto para luego poder instalarlo en el determinado dispositivo móvil que disponga de los requerimientos mínimos establecidos para su correcto funcionamiento.

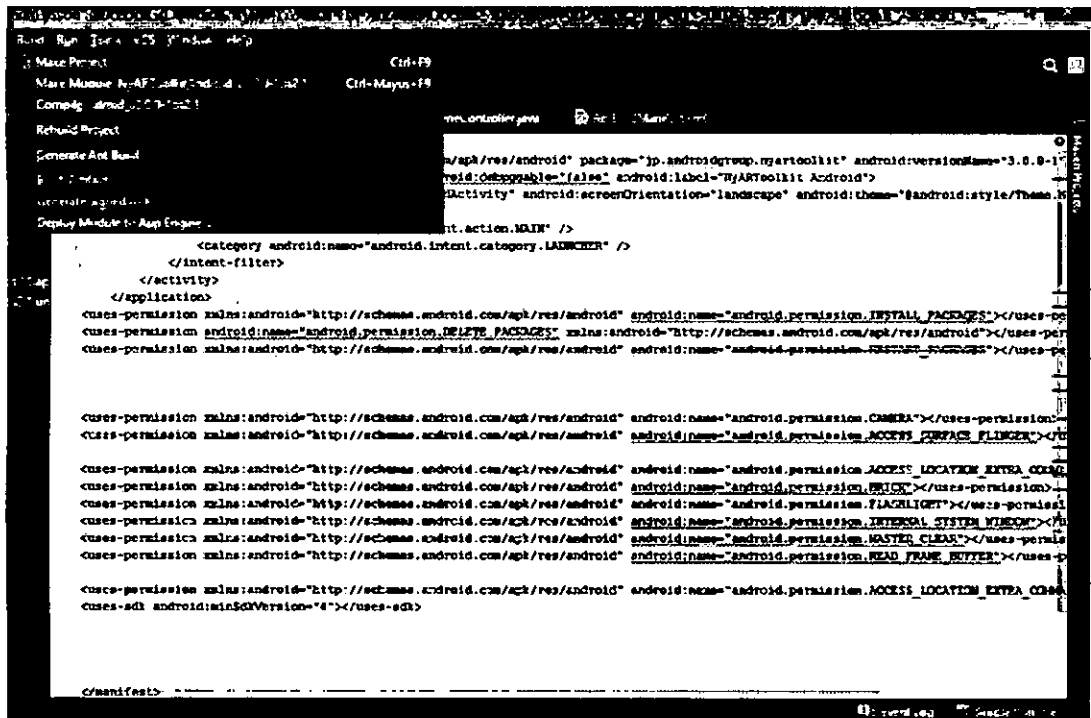


Figura 39. Exportación de la aplicación a extensión .apk

Fuente: Elaboración Propia

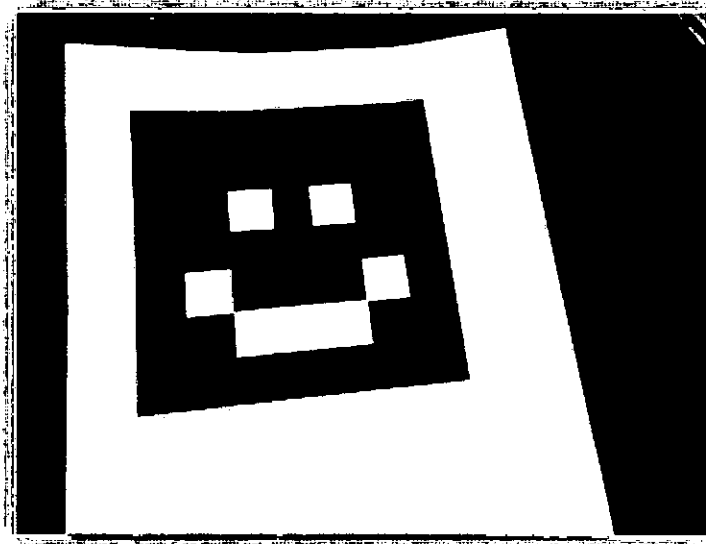
El archivo con la extensión .APK es el que se va a instalar. Para su instalación se realizan los siguientes pasos:

- Se copia el archivo con extensión .apk en el dispositivo móvil.
- Se busca el archivo con extensión .apk copiado en el dispositivo móvil y se ejecuta.
- Esperar hasta que se ejecute la instalación.
- Aceptar los permisos que aparecerán en pantalla para que la aplicación pueda instalarse correctamente.
- Una vez instalado probar y escanear un marcador previamente impreso, que sea conocido por la aplicación para poder utilizarla.



### V.1.7 Prueba con marcadores impresos

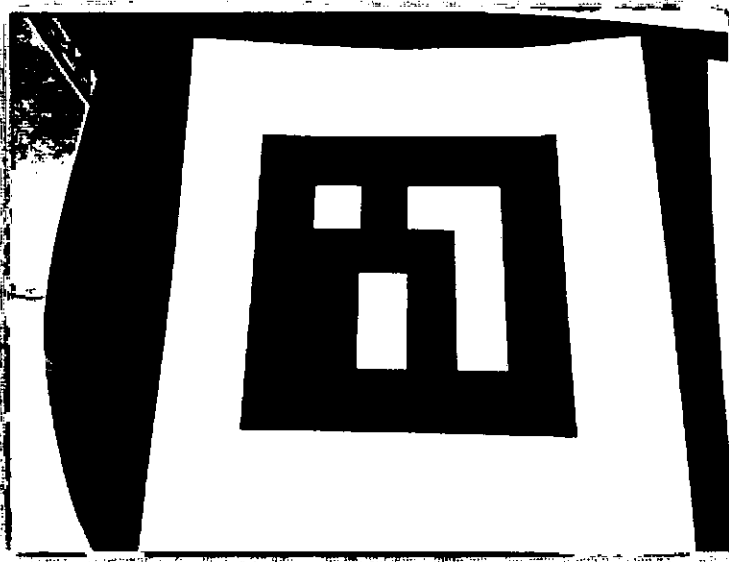
Una vez obtenidos los marcadores se imprimen y se colocan en una determinada distancia para luego escanearlos con la aplicación y realizar las pruebas de funcionamiento.



**Figura 40. Marcador1 impreso**

**Fuente: Elaboración Propia**

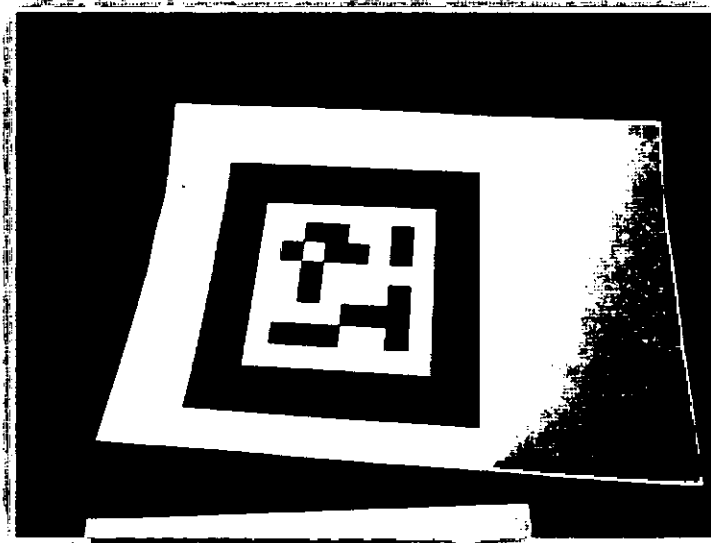
Este es el primer marcador impreso, el cual está asociado a la cama de madera de 2 plazas.



**Figura 41. Marcador 2 Impreso**

**Fuente: Elaboración Propia**

Este segundo marcador está vinculado al escritorio de madera para oficina.

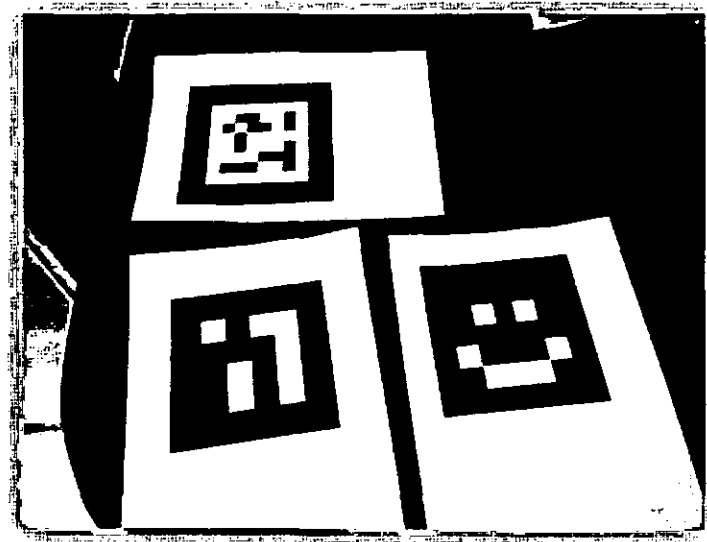


**Figura 42. Marcador 3 Impreso**

**Fuente: Elaboración propia**

Este tercer marcador está vinculado a la laptop Samsung core i3.



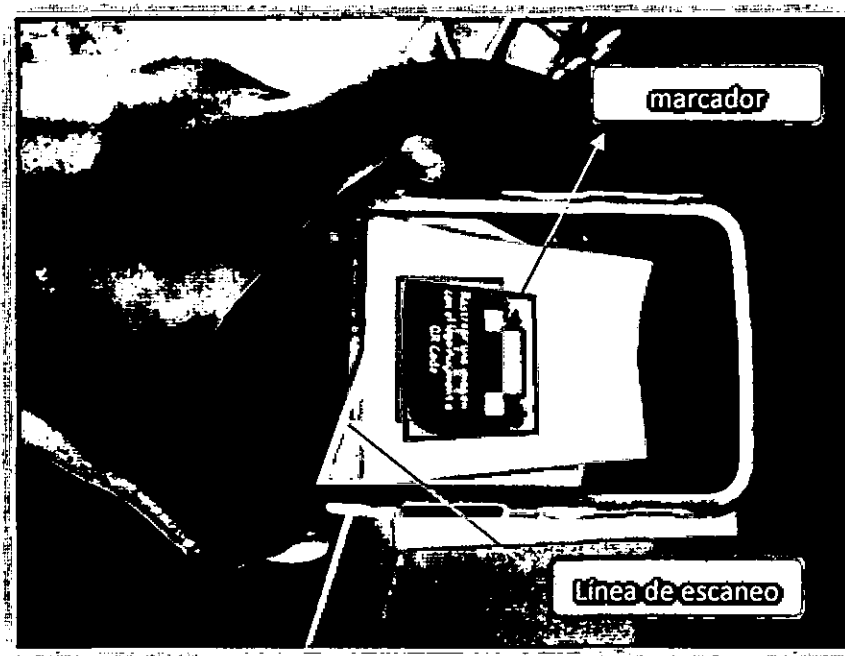


**Figura 43. Marcadores impresos**

**Fuente: Elaboración Propia**

Acá se muestra a los 3 marcadores impresos, que están listos para iniciar las pruebas.

Ahora se procede a escanear las imágenes con el dispositivo móvil, en este caso voy a utilizar un celular Samsung S5 con sistema operativo Android.



**Figura 44. Scaneo de Prueba en marcador 1**

**Fuente: Elaboración Propia**

En esta primera imagen se observa al dispositivo Smartphone iniciando el escaneo del marcador, en esta prueba se utiliza el marcador de la camade madera de 2 plazas.

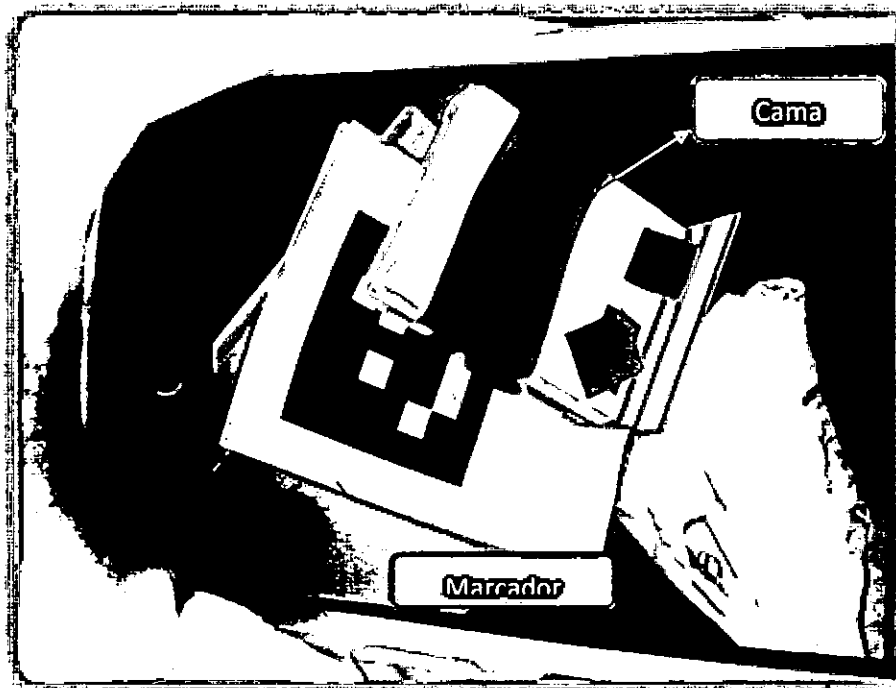
Dentro de la pantalla del celular se puede observar una línea de color rojo, la cual indica que está escaneando la imagen impresa, en este caso el marcador de la cama de 2 plazas.



**Figura 45.    Scaneo de Prueba en marcador 1 en proceso**

**Fuente: Elaboración Propia**

Cuando ya reconoce al marcador la aplicación se activa la realidad aumentada por medio de la comparación del marcador escaneado y el que se tiene guardado ya en memoria de la aplicación, de esta manera carga el objeto 3D vinculado a este marcador en este caso es la cama de madera de 2 plazas.



**Figura 46. Muestra objeto 3D con marcador**

**Fuente: Elaboración Propia**

En esta figura ya se observa que la activación de la realidad aumentada es exitosa ya que coincide el marcador escaneado con el de la aplicación y por consiguiente se muestra la imagen 3D, en este caso como se puede observar es la cama de madera de 2 plazas.



**Figura 47. Muestra objeto 3D con marcador 1**

**Fuente: Elaboración Propia**

Con lo que se obtuvo y se observó de la aplicación se pudo concluir que la prueba de funcionalidad de la aplicación se superó, ya que no se presentó ningún inconveniente durante la ejecución de la prueba con los marcadores impresos.

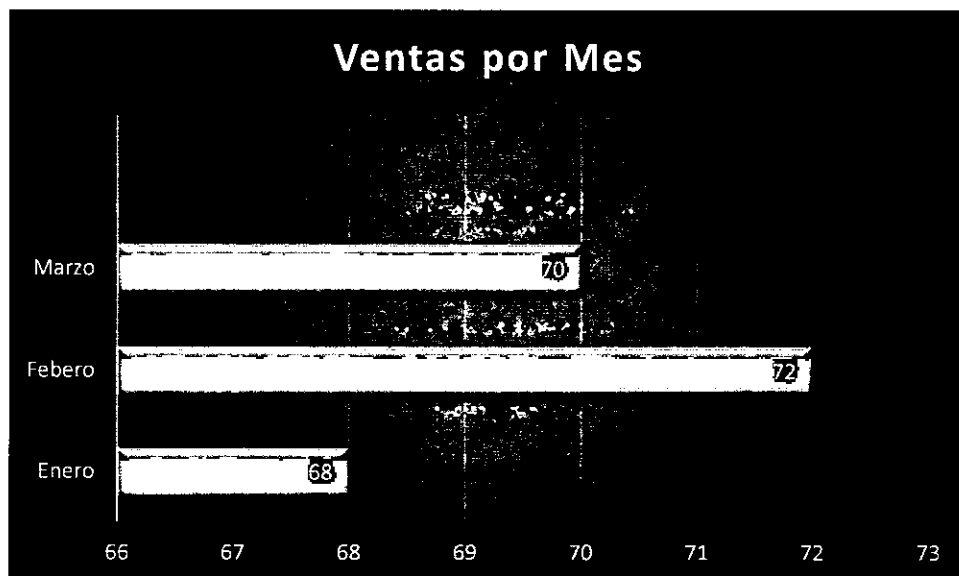
Una vez Realizadas las pruebas de funcionamiento se llevó la aplicación para que se ponga en funcionamiento dentro de la tienda importaciones Laban S.A.C sucursal Huancabamba, para poner en marcha la medición y del número de ventas con el uso de esta aplicación.



## V.2 CONSTATACIÓN DE RESULTADOS

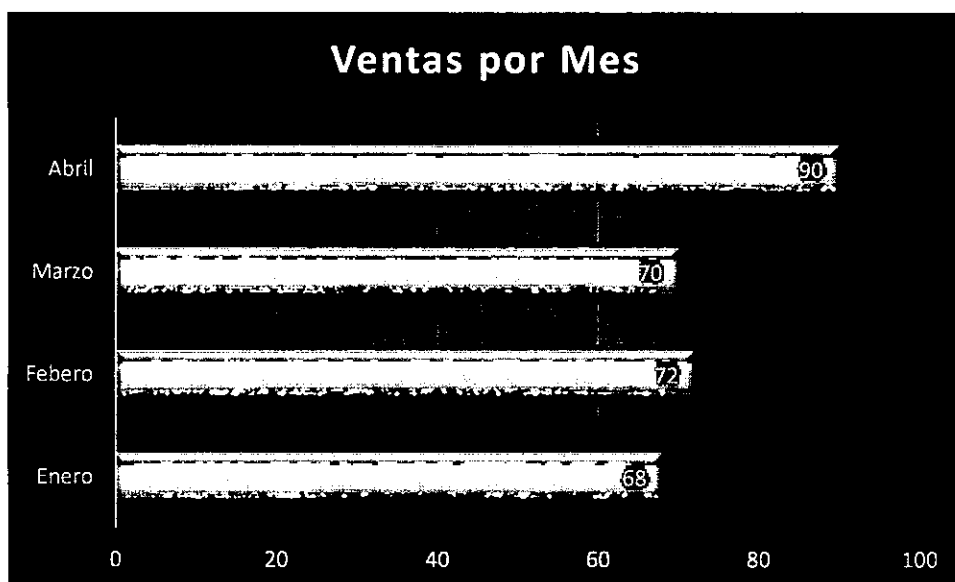
La constatación de resultados es la verificación y comprobación de los resultados obtenidos del número de ventas en la tienda Importaciones Laban S.A.C con respecto a la Hipótesis planteada en la Tesis.

En este primer cuadro se muestra el número de ventas de los meses anteriores antes de usar la aplicación. En el mes de enero se realizaron 68 ventas, en el mes siguiente 72, luego en el mes de marzo se realizaron 70 ventas.



**Figura 48.** Ventas de los meses de enero, febrero y marzo.

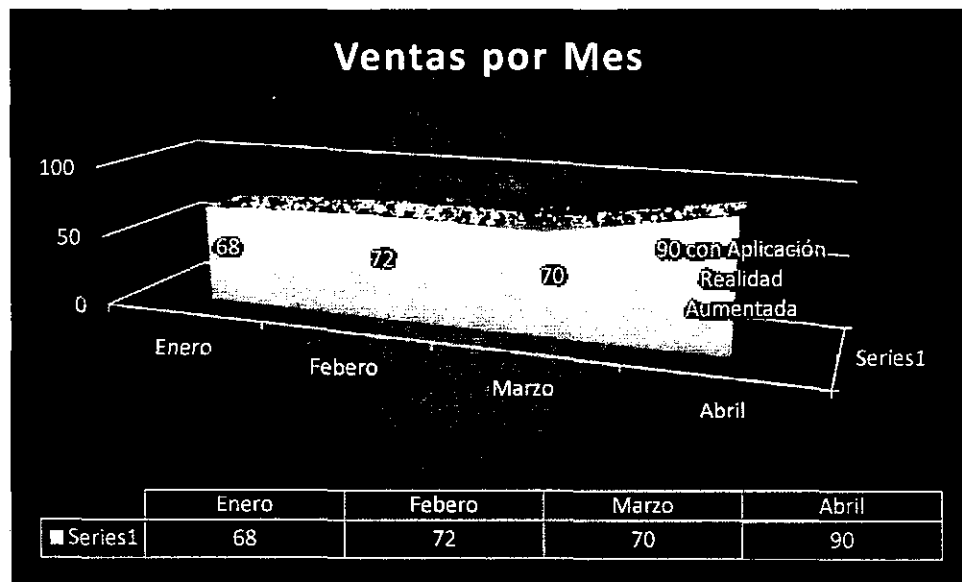
**Fuente:** Elaboración propia



**Figura 49. Número de ventas por mes [enero –abril]**

**Fuente: Elaboración Propia**

En el Cuadro anterior incluyo ya el mes de Abril en el cual ya se ha utilizado la aplicación en la tienda Importaciones Laban S.A.C sucursal Huancabamba para la muestra de los productos y así ayudar a la decisión rápida por parte de los clientes para comprar uno de esos tres productos en la tienda.



**Figura 50. Ventas por mes incluye [Enero-Abril]**

**Fuente: Elaboración Propia**

En este cuadro 3 muestro las ventas desde el mes de Enero hasta Abril donde se puede apreciar un incremento del 28.5% con respecto al número de ventas realizadas.

Promedio de ventas por mes 70

70 ventas → 100%

90 ventas → X

Aplicando regla de 3 simple:

$X = 90 \text{ ventas} \times 100\%$

-----  
70 ventas

**$X = 128.57\%$**





Con esto puedo comprobar que el uso de la aplicación influye positivamente para el incremento de las ventas y las ventas aumentaron un 28.57% respecto a los meses anteriores.

### **Hipótesis:**

H<sub>0</sub>: Las ventas de la empresa Importaciones Labán SAC- Huancabamba mejorarán con la implementación de la Aplicación basada en Realidad Aumentada.

**Por lo tanto la Hipótesis se acepta, con respecto a los resultados obtenidos.**

Dentro del uso de la aplicación de realidad aumentada en la tienda importaciones Laban S.A.C – Huancabamba, también se les preguntó a los clientes y personas en general que visitaron la tienda si estarían de acuerdo en comprar productos de la tienda con solo verlos en la aplicación de realidad aumentada y que opinaban de la aplicación.

Para esto se realizaron las siguientes preguntas:

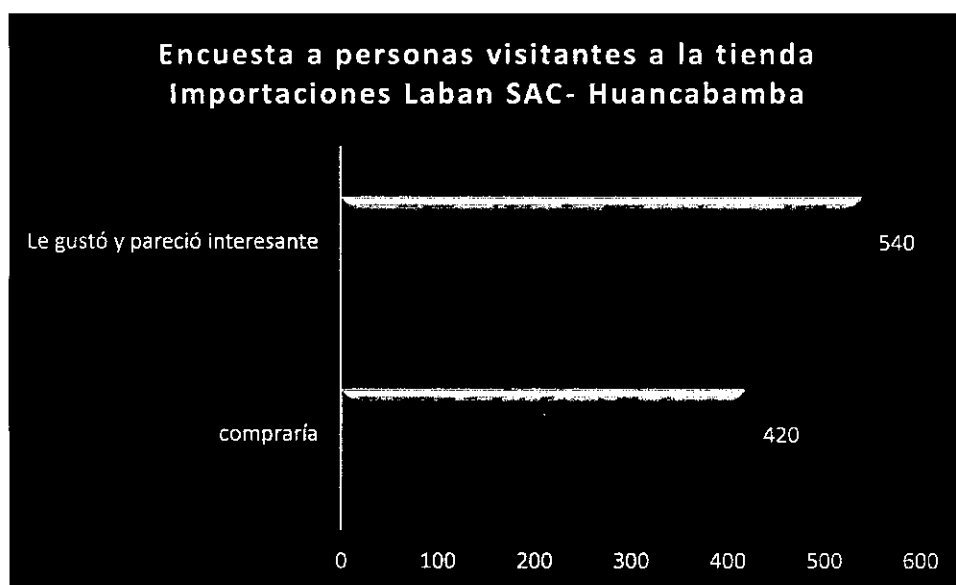
¿Usted compraría un producto con solo verlo en la aplicación?

¿Le gustó y pareció interesante la aplicación de Realidad Aumentada?



Estas preguntas se les realizaron a 600 personas en el mes de abril, en la tienda Importaciones Laban. S.A.C, y en el cual se obtuvieron los siguientes resultados.

El 70% de las personas que se les preguntó si compraría un producto con solo verlo en la aplicación y al 90% de las personas les gustó y les pareció interesante la aplicación.



**Figura 51. Encuesta a usuarios del mes de Abril**

**Fuente: Elaboración Propia**

Con estas dos preguntas realizadas a estas 600 personas realizadas durante el mes de abril, se pudo obtener como resultado que la aplicación es aceptada por la mayoría de las personas que la han visto de cerca y que trae beneficios positivos para la empresa.



## CONCLUSIONES

- El buen uso de la Tecnología en los negocios trae mejores resultados ya que en muchos negocios de diferentes lugares del mundo se utiliza y muestra gran aceptación por parte de los usuarios. Reacción del usuario.
- Con la aplicación de realidad Aumentada la toma de decisiones en los clientes es más efectiva ya que los hace percibir en tiempo real la simulación de un determinado objeto en un determinado espacio donde se encuentran o donde ellos quisieran que esté.
- El número de ventas con el uso de la aplicación mejoraron más del 20% con respecto a los meses anteriores que no se utilizó la aplicación en Importaciones Laban S.A.C.
- La metodología Scrum ayudó a estimar un tiempo de referencia para el desarrollo del proyecto, pero este tiempo estimado no fue preciso ya que de acuerdo a los requerimientos del usuario final hubieron cambios los cuales demandaron de más tiempo modificando el tiempo estimado en primera instancia.
- La aplicación es muy bien aceptada por las personas que vieron de cerca la aplicación ya que el 70% de estas personas compraría con solo ver el producto mediante la aplicación y a más del 89% le gustó la aplicación lo cual es muy positivo para las ventas en la empresa.



## RECOMENDACIONES

- Saber utilizar Herramientas de diseño 3D para que nos facilite el uso y manejo de estos objetos ya que no siempre se podrá obtener estos objetos ya diseñados en internet.
- Agregar un módulo de emisión de lista de productos seleccionados para compra dentro de la tienda de modo online, agregar un módulo de enlace remoto para envío del pedido a la empresa y retorno de la cotización, agregar un módulo de impresión en impresora portátil si es que se piensa seguir usando marcadores.
- Desvincular la dependencia de marcadores para poder visualizar los objetos 3D ya que esto limita la portabilidad de la aplicación a tener siempre marcadores para poder ver los objetos 3D en cualquier lugar que nosotros deseemos.
- Implementar también para IOS ya que el mercado más amplio en los dispositivos móviles lo ocupa Android e IOS



## BIBLIOGRAFÍA

### TESIS EN LA WEB

SALAZAR ALVAREZ, IVÁN ANDRÉS; (2013). “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA INFORMACIÓN TURÍSTICA BASADO EN REALIDAD AUMENTADA”. (Tesis para obtener el título de Ingeniero en Telecomunicaciones). Pontificia Universidad Católica del Perú. Recuperado de:

[http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4647/SALAZAR\\_IVAN\\_REALIDAD\\_AUMENTADA.pdf?sequence=1](http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4647/SALAZAR_IVAN_REALIDAD_AUMENTADA.pdf?sequence=1)

Recio Lecuona, Francisco; (2012). “DESARROLLO EN ANDROID PARA APLICACIONES DE REALIDAD AUMENTADA”. (Tesis para obtener el título de ingeniero en Informática). Universidad del País Vasco. Recuperado de:

[https://addi.ehu.es/bitstream/10810/9025/4/Memoria\\_Francisco\\_Recio.pdf](https://addi.ehu.es/bitstream/10810/9025/4/Memoria_Francisco_Recio.pdf)



## ARTICULOS EN INTERNET

### Bibliografía

3D, A. (13 de 01 de 2015). *Archive 3D*. Obtenido de Archive 3D:  
<http://archive3d.net/?a=download&id=f1ca7105>

Akanksha. (25 de 09 de 2014). *DeveloperIQ*. Obtenido de DeveloperIQ:  
<http://developeriq.in/articles/2014/sep/25/developing-android-augmented-reality-applications/>

Bautista Rojas, M. E., & Archila Díaz, M. F. (2011). TAREAS FUNDAMENTALES EN LA REALIDAD AUMENTADA, UN NUEVO. *Revista Colombiana de Tecnologías de Avanzada*, 57. Obtenido de [http://www.unipamplona.edu.co/unipamplona/portallIG/home\\_40/recursos/04\\_v19\\_24/revista\\_19/09022012/09.pdf](http://www.unipamplona.edu.co/unipamplona/portallIG/home_40/recursos/04_v19_24/revista_19/09022012/09.pdf)

CNET. (22 de Enero de 2015). *Microsoft salta a la realidad aumentada con los HoloLens*. Obtenido de El Comercio:  
<http://elcomercio.pe/tecnologia/inventos/microsoft-salta-realidad-aumentada-hololens-noticia-1786253>

Dubretic, M. (25 de Abril de 2014). *iOS vs Android vs Windows: La batalla de los sistemas operativos móviles*. Obtenido de Udemý:  
<https://blog.udemy.com/es/ios-vs-android-vs-windows-la-batalla-de-los-sistemas-operativos-moviles/>

Icescrum. (13 de 01 de 2015). *Icescrum*. Obtenido de Icescrum:  
<http://www.icescrum.org/>

Innovae. (15 de Setiembre de 2013). *Realidad Aumentada*. Obtenido de Innovae Augmented Reality: <http://realidadaumentada.info/realidad-aumentada/>

Innovae. (15 de Setiembre de 2013). *Tecnología*. Obtenido de Innovae Augmented Reality: <http://realidadaumentada.info/tecnologia/>



Isabel Valencia. (20 de 12 de 2014). *Android Lollipop vs iOS 8 - La batalla de sistemas operativos*. Obtenido de ANDROIDPIT: <http://www.androidpit.es/android-lollipop-vs-ios-8>

Javier Guadiana. (31 de Julio de 2014). *Android es líder mundial en ventas de smartphones con un 85% de cuota de mercado*. Obtenido de Reason Why: <http://www.reasonwhy.es/actualidad/empresa/android-es-lider-mundial-en-ventas-de-smartphones-con-un-85-de-cuota-de-mercado>

kunagi. (12 de 01 de 2015). *kunagi*. Obtenido de kunagi: <http://kunagi.org/>

KZGUNEA, E.-L. (20 de Junio de 2013). *Realidad Aumentada*. Obtenido de PLATAFORMA E-LEARNING KZGUNEA: <http://e-forma.kzgunea.net/mod/book/view.php?id=393>

M RUTH GAMERO. (9 de Setiembre de 2011). *Realidad Aumentada: una nueva lente para ver el mundo (I)*. Obtenido de La Cofa: <http://www.lacofa.es/blog/2011/09/09/realidad-aumentada-una-nueva-lente-para-ver-el-mundo-i/>

Orozco, D. (1 de Junio de 2011). *Definición de Android*. Obtenido de CONCEPTODEFINICION.DE: <http://conceptodefinicion.de/android/>

PangoScrum. (13 de 01 de 2015). *PangoScrum*. Obtenido de PangoScrum: <http://pangoscrum.com/>

Pedro Gutiérrez. (7 de Enero de 2011). *IOS*. Obtenido de SO MOVIL: <http://so-movil.xitrus.es/ios/index.html>

Protalinski, E. (8 de 12 de 2014). *VB news*. Obtenido de VB news: <http://venturebeat.com/2014/12/08/google-releases-android-studio-1-0-the-first-stable-version-of-its-ide/>

Salazar Alvares, I. A. (5 de Febrero de 2013). *Realidad Aumentada. DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA INFORMACIÓN TURÍSTICA BASADO EN REALIDAD AUMENTADA*, 33-36. Lima, Lima, Perú: Pontificia Universidad Católica del Perú. Obtenido de



[http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4647/SALAZAR\\_IVAN\\_REALIDAD\\_AUMENTADA.pdf?sequence=1](http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4647/SALAZAR_IVAN_REALIDAD_AUMENTADA.pdf?sequence=1)

Samsung. (13 de 03 de 2014). *Smart GSM*. Obtenido de Smart GSM:  
<http://www.smart-gsm.com/moviles/samsung-galaxy-s3>

Scrumdo. (13 de 01 de 2015). *Scrumdo*. Obtenido de Scrumdo:  
<https://app.scrumdo.com>

Sprintometer. (13 de 01 de 2015). *Sprintometer*. Obtenido de Sprintometer: <http://sprintometer.com/>

Vegas, E. (06 de 09 de 2014). *Emiliusvgs*. Obtenido de Emiliusvgs:  
<http://emiliusvgs.com/2014/09/06/tutorial-basico-realidad-aumentada-unity/>





## ANEXOS

### Anexo 1:

#### TUTORIAL BÁSICO DE REALIDAD AUMENTADA CON UNITY

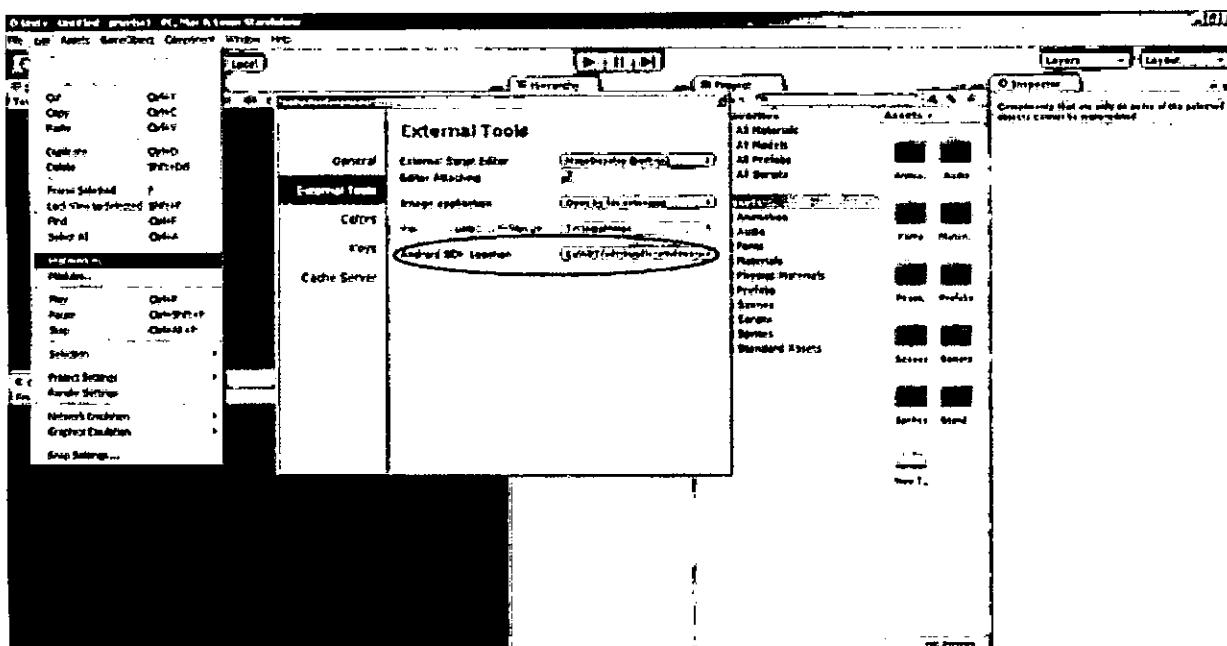
Fuente: (Vegas, 2014)

He hablado muchas cosas sobre Unity3D. Todo comenzó en el 2011 cuando buscaba un software para hacer videojuegos. Desde allí he tenido la fascinación de profundizar más en armar y programar algo de manera profesional. Lamentablemente no he podido generar algo tan PRO como quisiera. Sin embargo trabajé y experimenté todo lo posible que se me ocurriera. Lo que más he experimentado, aparte de crear terrains es proyectar **Realidad Aumentada**. Si bien trabajé diferentes aspectos de la realidad aumentada no he mostrado un tutorial básico (paso por paso) así que vamos a escribir un artículo interesante:

1) Descargar la última versión de Unity. Existe mejoras en todas las versiones y características por ello avanzamos siempre hacia las nuevas descargas. Ahora estamos en Unity 4.5

2) Es necesario tener instalado el **SDK de Android**. Si vas a exportar en unity3D es obvio que debes tener en cuenta que debes generar un APK (app). Lo cual para poder hacerlo debes tener configurado tu interfaz.

Para hacer ello debes entrar a EDIT —> Preference —> External Tools->android



3) Utilizaremos vuforia, su **versión de extensión**. En la actualidad estamos en la versión 3. El SDK tiene su propio instalador por lo que no te preocupes por el tema de coordinar códigos, el set-up y todo el enrolllo que aveces puede existir. Si vuforia con eclipse es un dolor de cabeza; vuforia con unity3D es como instalar un software: dos clics y listo.



Android

IOS

Unity Extension

## Unity Extension - Vuforia v3.0

The Vuforia™ Unity Extension allows you to build vision-based augmented reality applications for building a cross-platform game engine.

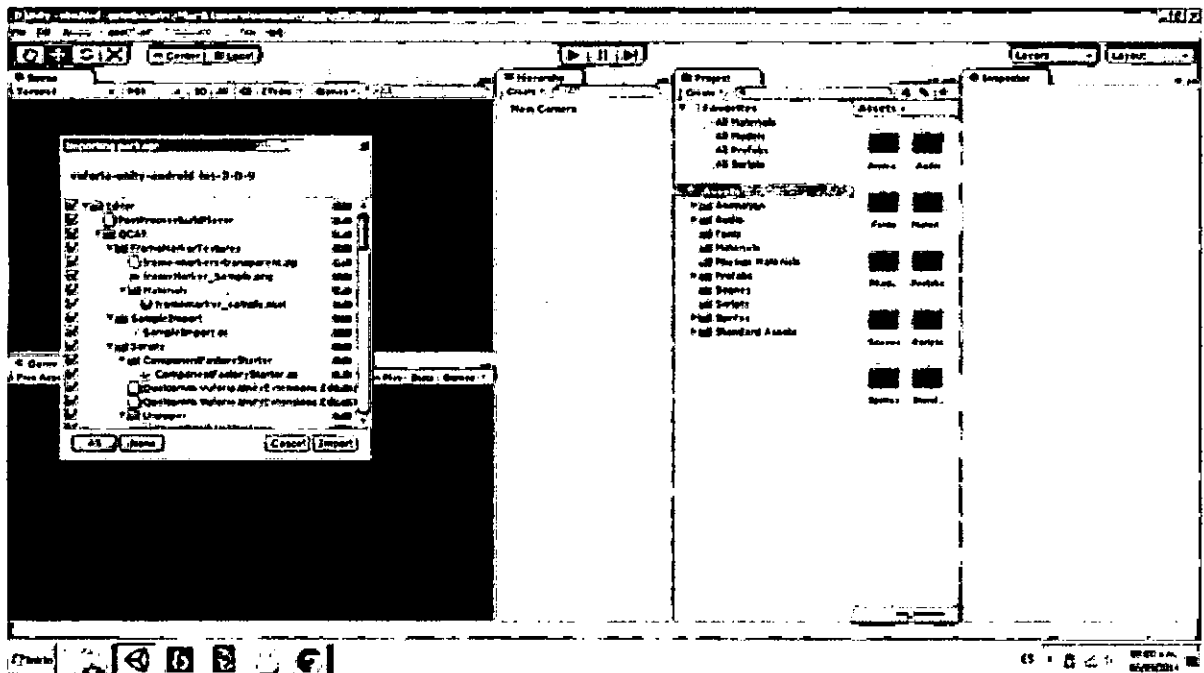
Download Unity Extension 3.0.9 for Android & IOS (18.36 MB)

### Getting Started - Installing the Vuforia SDK

The Vuforia AR Extension for Unity allows developers to build AR apps with the popular cross-platform game engine - Unity. Developers must first install Unity for Android and/or Unity for IOS from Unity Technologies. The Vuforia AR Extension is compatible with both Unity Standard and Unity Pro. Visit the Unity website for further information about Unity.

Developers who have already installed Unity can download the Vuforia AR Extension from Downloads or directly from the Unity Asset Store. See Step 1: Installing the Unity Extension for further details on each of these options.

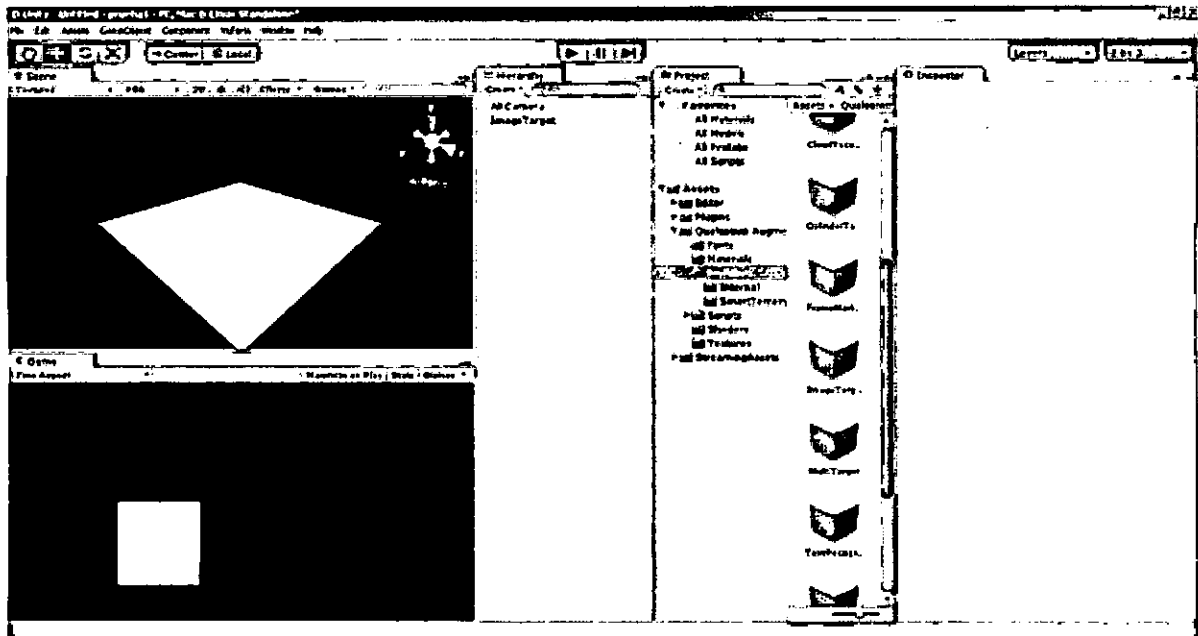
Como habrás visto, instalar fue importar el package. Le damos a todo Ok y aparecerá en la carpeta



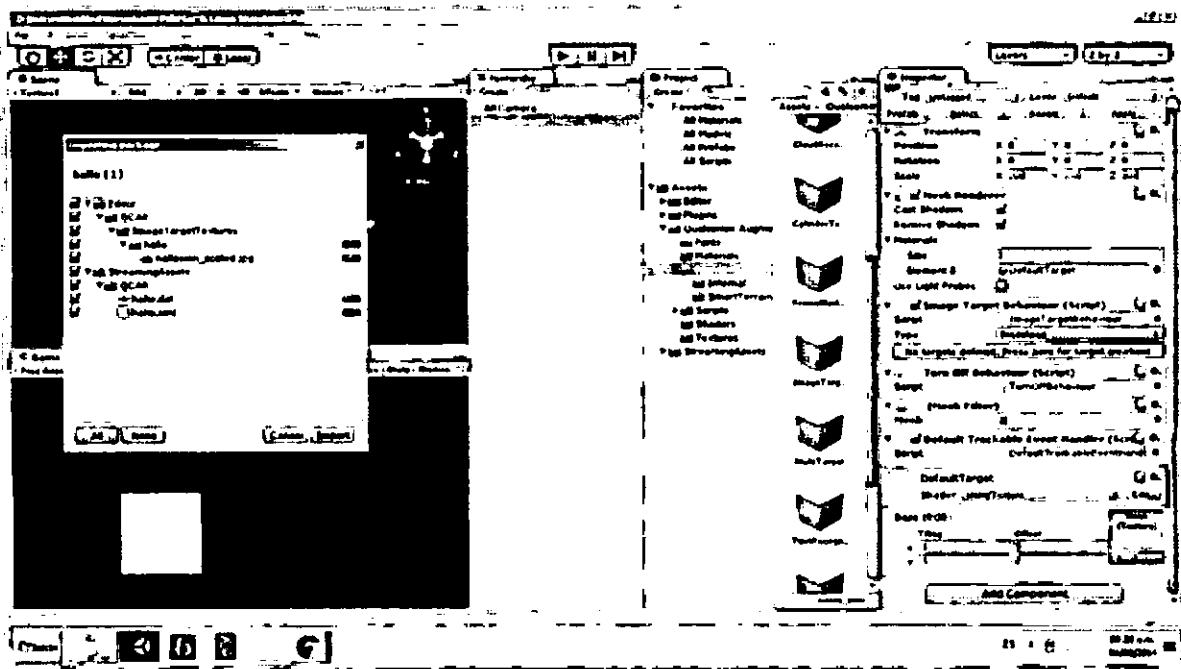
4) En la carpeta Assets, aparecerá todo nuestro material instalado. Tenemos que buscar Prefabs para hallar todos los ejemplos para ser ejecutados. Por hoy vamos iniciar con **Image\_Targets**. Este ejemplo se caracteriza por hacer correr el ejemplo simple de un marcador de color y un objeto 3D.



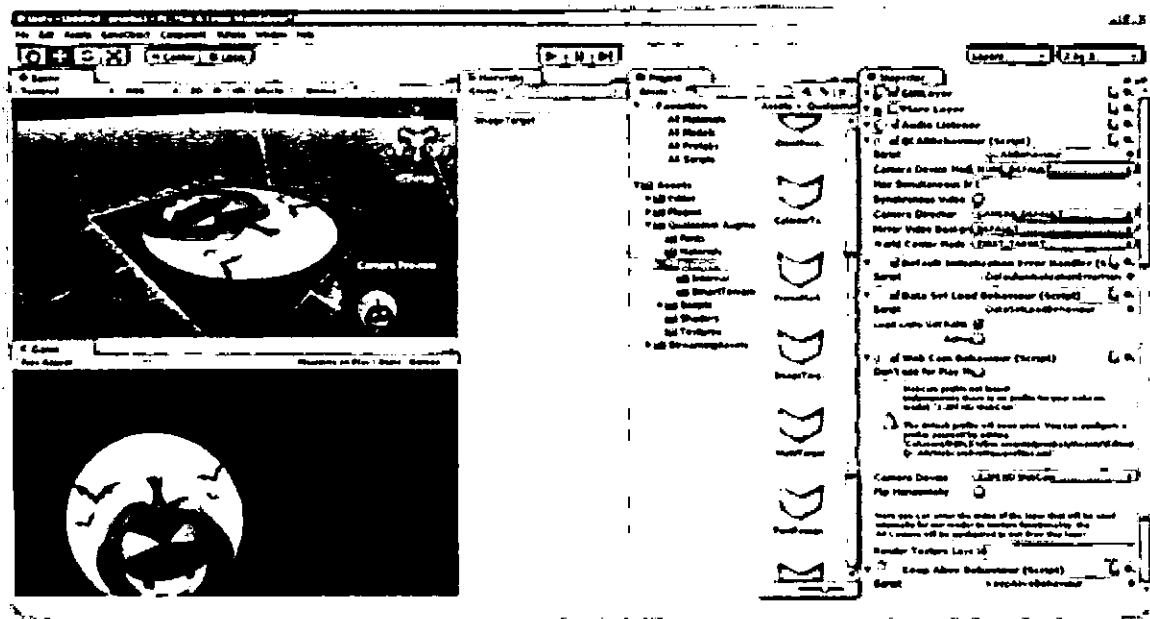
5) Limpiemos el lugar de trabajo. Por el momento quitamos el “main camera” para evitar posible doble sonido o interrupción de la visualización. Arrastramos AR CAMARA, y luego IMAGE TARGET. Debemos tener ambos archivos uno tras otro y no como subordinado.



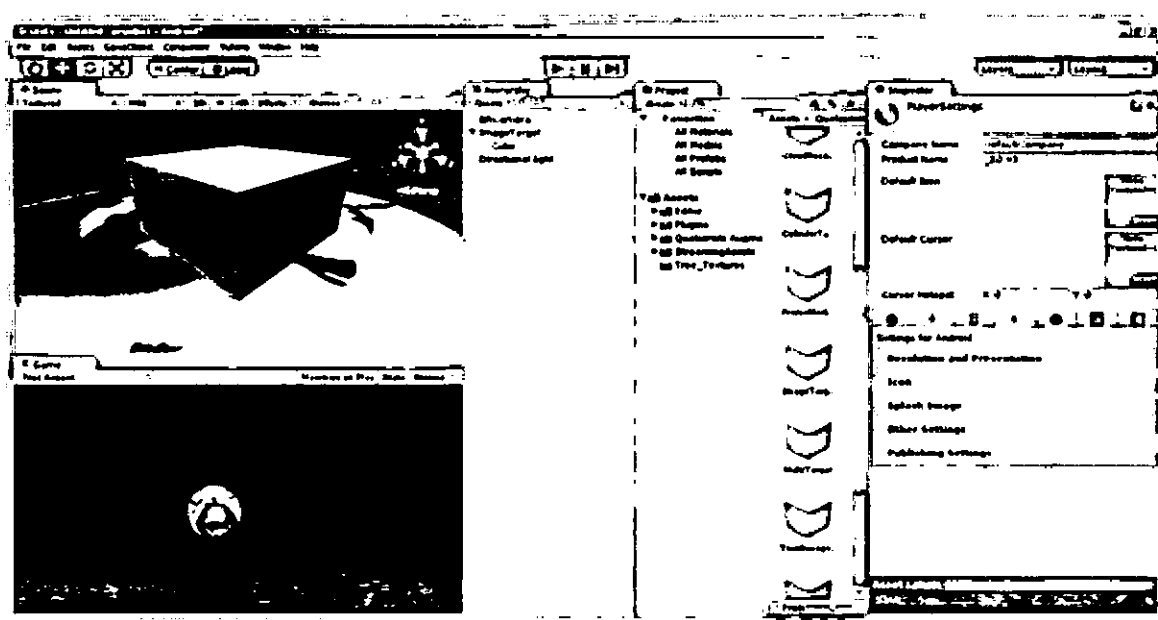
6) Es momento de agregar nuestro marcador. Nos situamos en **Image Target** y en las opciones que aparecen en inspector encontramos un apartado llamado **Imagen Target Behaviour**. Veremos que aparece el mensaje “NO TARGETS DEFINE...”. Le damos clic y nos llevará al target manager de vuforia. Subimos una foto que sea trackeable y cuando tenemos todo listo realizamos la descarga en modo Unity editor e importamos.



En la misma opción de Image Target Behaviour ya aparece otras opciones. En data set aparece cuales son mis archivos trackeables. Por ahora aparece mi archivo llamado "Halo". Con tal solo darle clic al nombre ya cambia el marcador blanco a la imagen correspondiente. Ya hicimos set-up allí, ahora vamos a AR CAMERA. Como ya integramos el marcador, en la opción de Data Set Load Behaviour marcamos con check Load Data Set "name"... y también active.



Con esto solo queda agregar cualquier textura. Por ejemplo un cubo o una esfera. Recuerda que todo material debe estar dentro de Image Target. Si lo colocas fuera se verá mal pues no estarás dando el mensaje de lectura para el marcador. Finalmente para personalizar el trabajo debes ir a File -> Build Setting -> Android -> Player Setting. En este espacio podemos cambiar los ícones, los splash, nombres de compañía, etc. El resultado será algo sencillo pero abre las puertas a desarrollar cosas mucho más potentes.





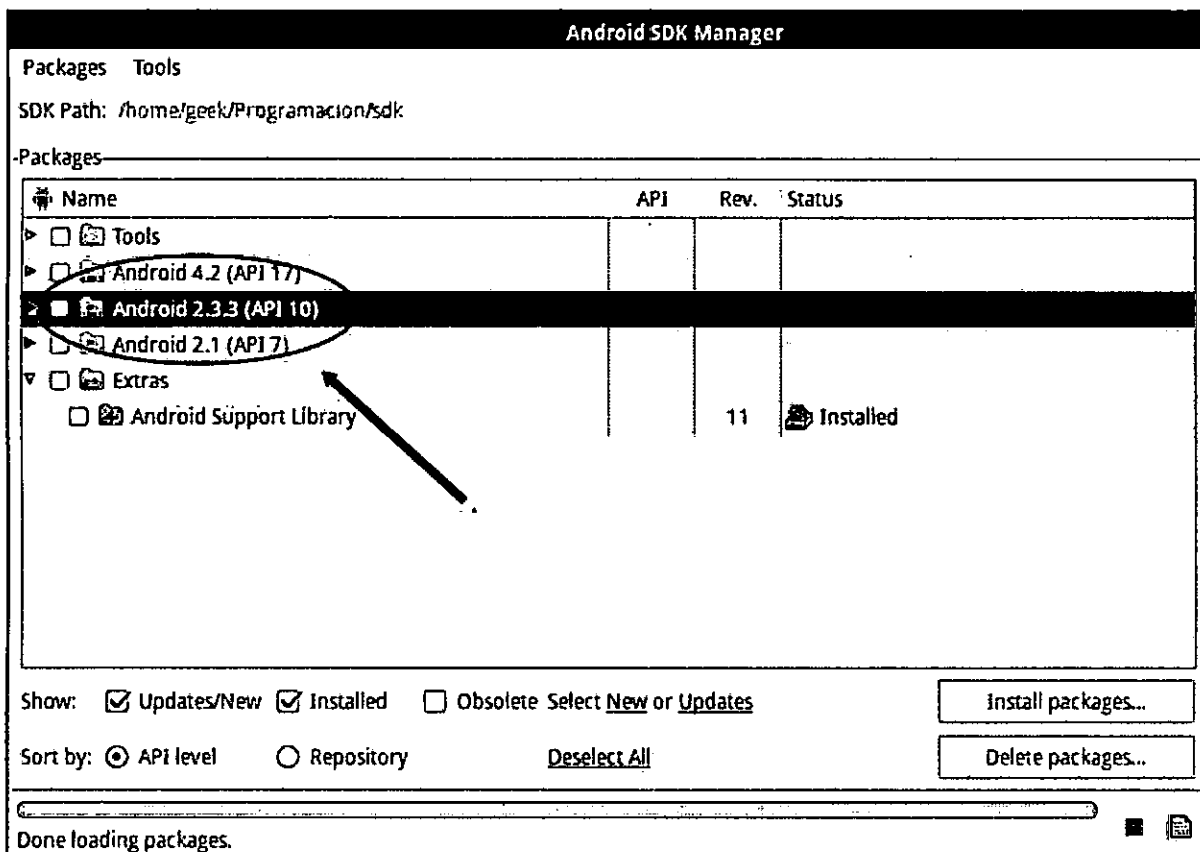
## Anexo 2:

### Tutorial de realidad aumentada para Android

- Misión: Tener documentación de nyartoolkit para nuestra comunidad en habla hispana.

**Ryu Iizuka** (creador de Nyartoolkit) tiene demasiados proyectos y poco tiempo para documentar. Por eso estamos nosotros. Investigamos y publicamos resultados.

Bueno pues a petición de Emilio, ya que tampoco encontré documentación sobre nyartoolkit, me decidí a aportar un poco de lo que aprendí al tratar de hacer realidad aumentada con `.md2` y **NyArToolKit for Android**. Para empezar este tutorial supongo que ya tienes instalado Android Bundle con la API de android de la version 2.3.2. Es obvio que usaremos eclipse para trabajar.





Debes saber que el modelo que quieras cargar será creado con **Blender 2.49 o 2.47**. Estas son versiones antiguas pero recordemos que la extensión **.md2** también es muy antigua y el algoritmo para cargarlo también lo es. Así que debes aprender primero a usar esa versión de Blender y debes saber como crear un modelo y exportar su textura a un solo archivo o de otra manera no funciona, por ejemplo del video de Emilio de exportar el cubo con 1 sola textura podría funcionar. Ahora cuando nosotros usamos la librería 'Nyartoolkit para processing' solo usamos 1 librería, pero cuando usamos Ny ArToolKit For Android usamos:

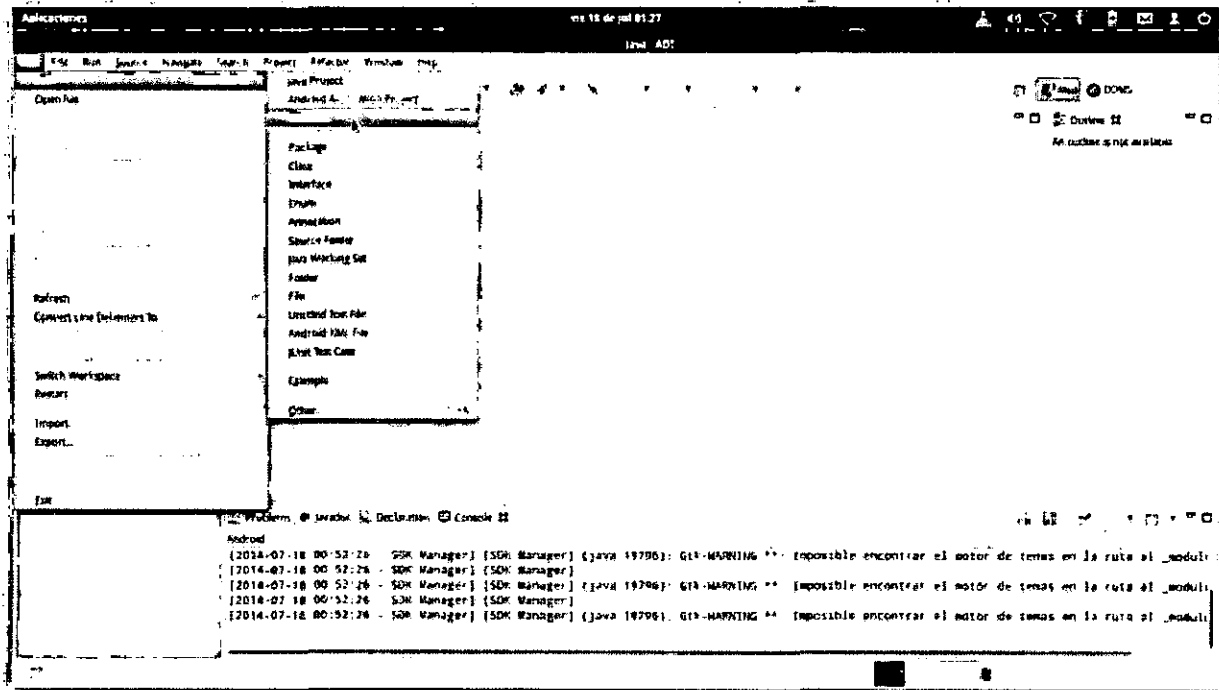
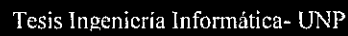
- NyArToolKit For Android
- Min3d

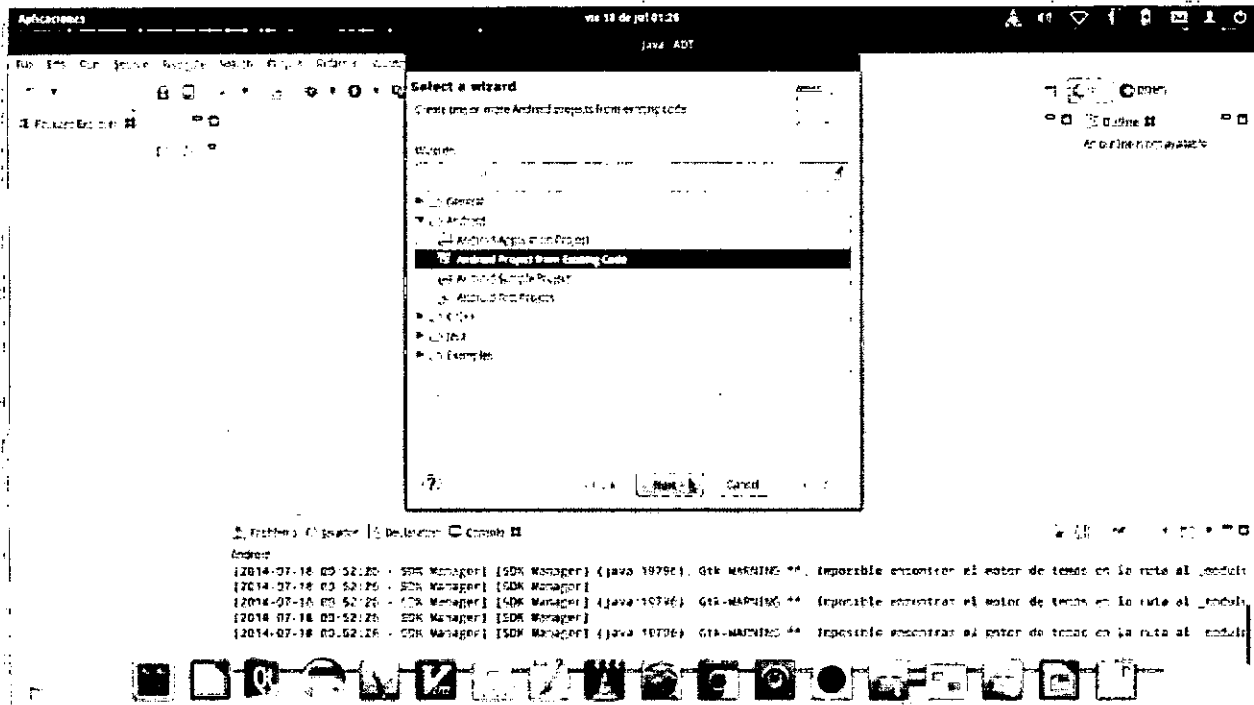
Los algoritmos de 'nyartoolkit for android' solo sirve para hacer realidad aumentada y el de min3d es para cargar objetos 3d ya sean **.md2** **.3ds** y **.obj** pero el ejemplo que ha hecho el autor solo está diseñado para que se puedan cargar los **.md2** y si llegas a tratar de cargar un **.3ds** o un **.obj** jamas te mostrara nada (al menos a mi no me funciona).

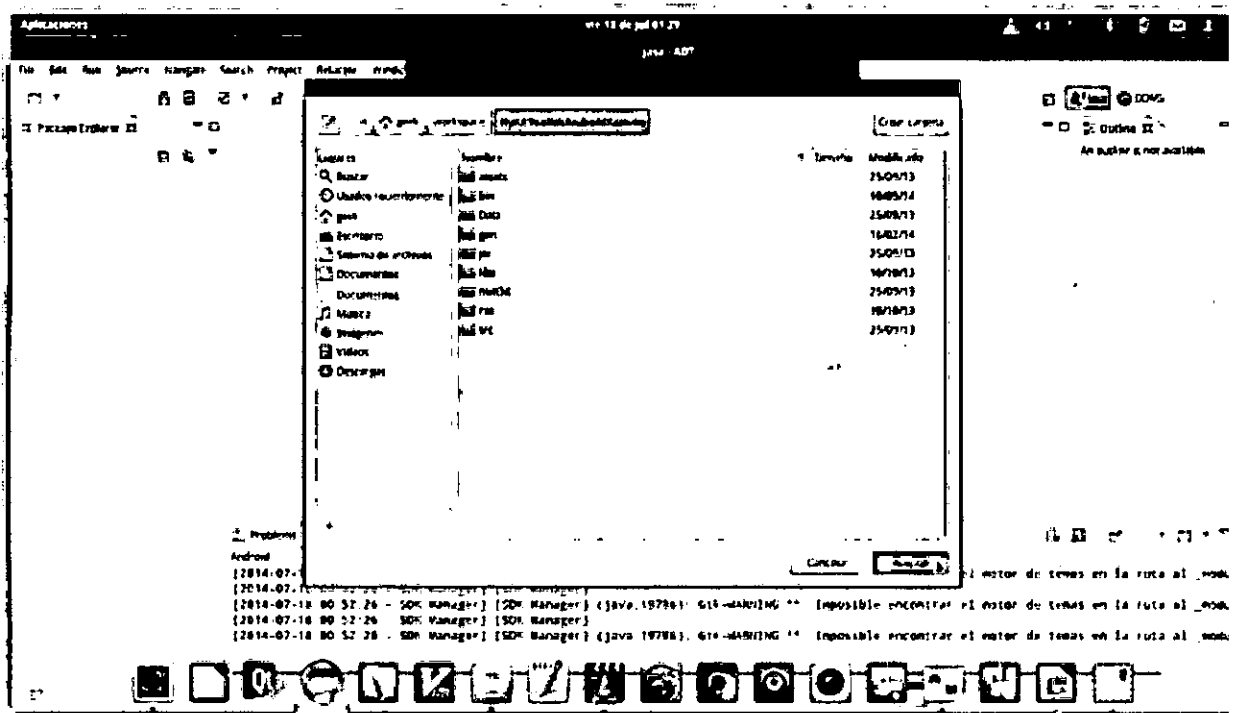
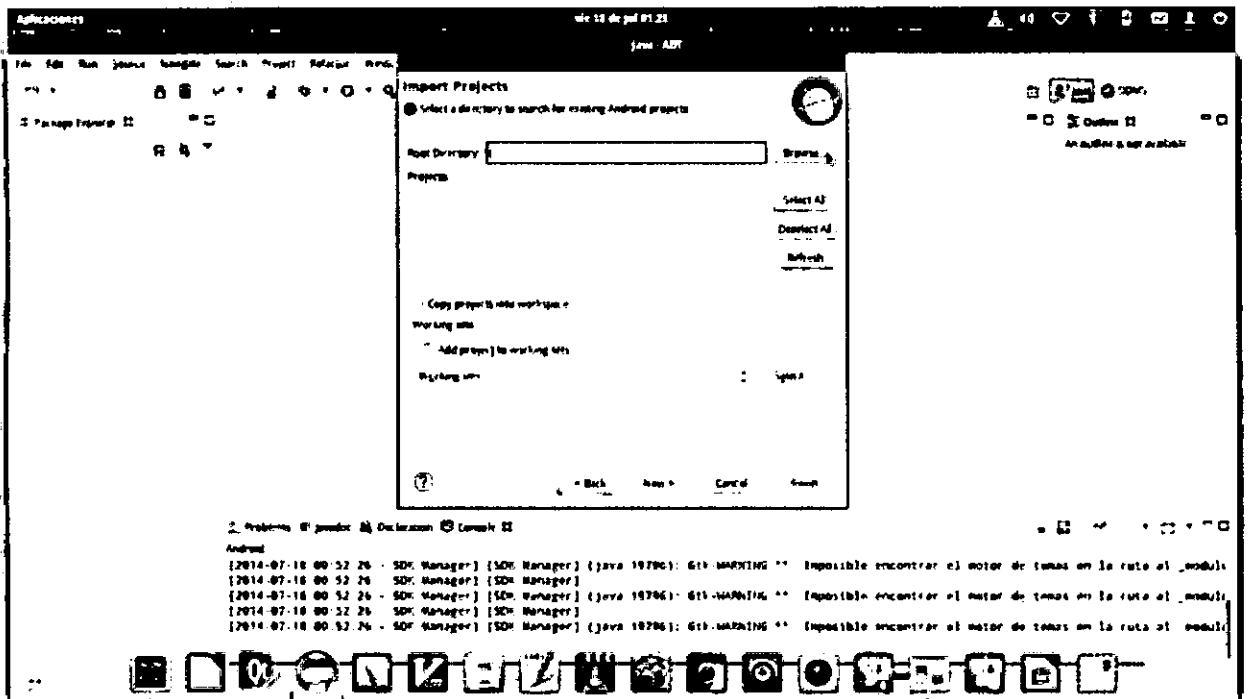
Ya tenemos todo lo necesario para hacer el set-up: Programa, versión de android, archivo descargable y la versión de blender. Ahora vamos a comenzar a programar!

Para este punto damos por sobreentendido que has tenido cierto conocimiento de eclipse y la manera de cómo trabaja android con su SDK.





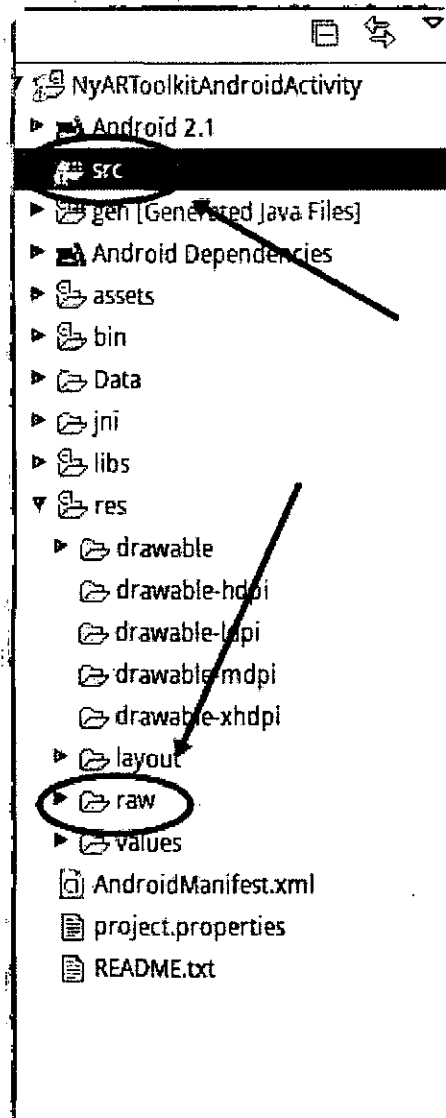




Si pudiste importar correctamente el proyecto debe aparecer algo así. Si te has equivocado en algo, intenta nuevamente. Al importar archivos por lo general tienen que saltar ciertas alertas en forma de dibujo. Mira este gráfico y date cuenta que el SRC sale



un signo de alerta y los assets, bin presenta un dibujo encima de carpeta. Al importar se agrega documentación y archivos que se deben modificar o tener en cuenta, por ello el programa lo resalta.



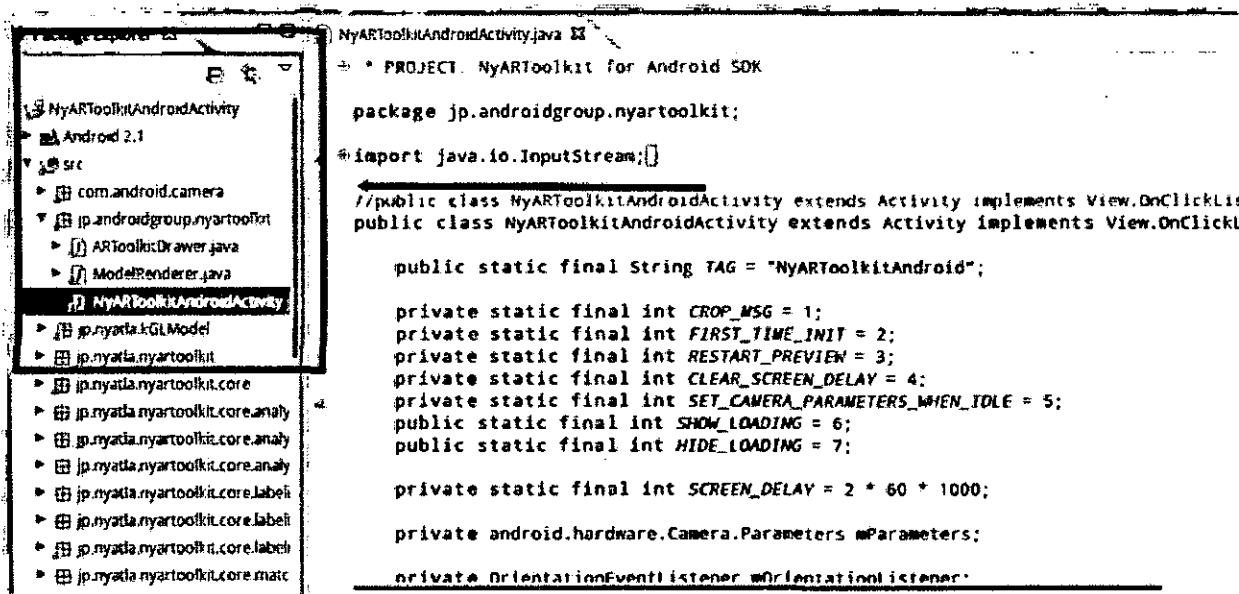


Ahora obtenemos los siguientes directorios:

- **src:** Es la carpeta en donde se encuentra todo el código necesario.
- **raw:** Es en donde pondremos nuestros modelos .md2



Ahora veamos el código que necesitamos modificar y este se ubica en NyArToolkitActivity.



El código de Nyartoolkitandroidactivity encierra todo el trabajo que podemos armar. Si bien está algo desfazado, es necesario utilizarlo pues sigue siendo un referente en la programación de realidad aumentada.



```
NyARToolboxAndroidActivity.java 21
/**
 * @see min3d.interfaces.ISceneController#initScene()
 */
public void initScene()
{
    scene.lights().add(new Light());
    scene.camera().frustum.zFar(10000.0f);
    //scene.camera().frustum.shortSideLength(0.77f);

    IParser parser;
    AnimationObject3d animationObject3d = null;

    parser = Parser.createParser(Parser.Type.MD2,
        getResources(), "jp.androidgroup.nyartoolkit:raw/droid", false);
    parser.parse();

    animationObject3d = parser.getParsedAnimationObject();
    animationObject3d.rotation().z = -90.0f;
    animationObject3d.scale().x = animationObject3d.scale().y = animationObject3d.scale().z = 1.0f;
    scene.addChild(animationObject3d);
    animationObject3d.setFps(30);

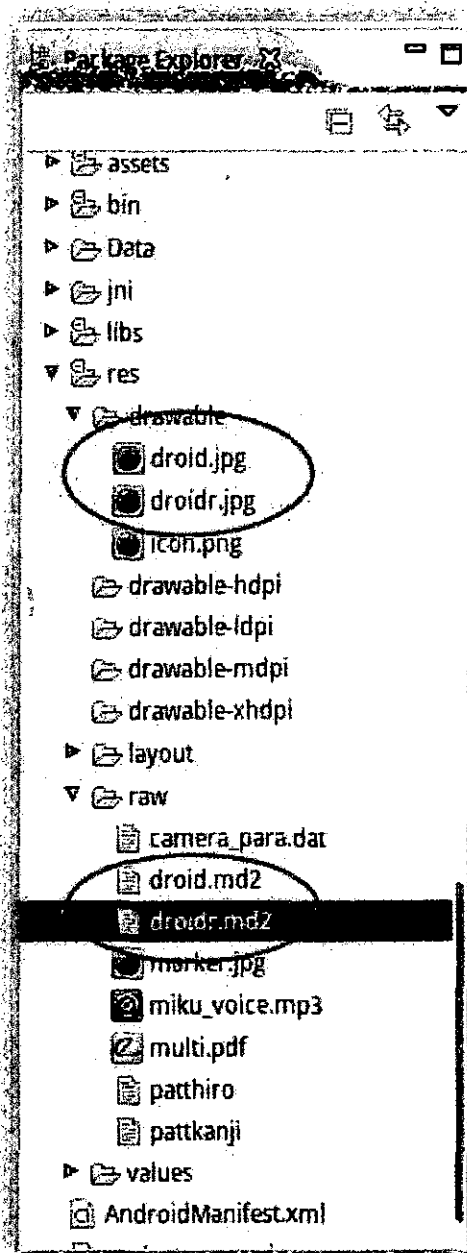
    parser = Parser.createParser(Parser.Type.MD2,
        getResources(), "jp.androidgroup.nyartoolkit:raw/droidr", false);
    parser.parse();

    animationObject3d = parser.getParsedAnimationObject();
    animationObject3d.rotation().z = -90.0f;
    animationObject3d.scale().x = animationObject3d.scale().y = animationObject3d.scale().z = 1.0f;
    scene.addChild(animationObject3d);
    animationObject3d.setFps(90);
}

/**
 * All manipulation of scene and Object3D instance properties should go here.
 */
```



Droid y droidr son los nombre de los archivos .md2 que estan en la carpeta raw.



Ahora en la carpeta drawable debemos meter la textura de cada uno de los modelos que queramos cargar es importante que las texturas o archivos png o jpg sean con el mismo nombre que el de los modelos. Osea si mi modelo se llama mimodelo.md2, la textura se debe llamar mimodelo.png

Patthiro y pattkanji son los 2 marker por defecto pero podemos cambiarlos sin ningún problema. Ahora por si queremos agregar nuestros propios markes solo faltaría hacerlos con el marker generator que se encuentra en la web y guardarlos la carpeta drawable y modificar esta linea de código.



```
private void initializeGLSurfaceView() {  
    // init ARToolkit.  
    if (artToolkitDrawer == null) {  
        InputStream camePara = getResources().openRawResource(R.raw.camePara);  
        int[] width = new int[2];  
        for (int i = 0; i < 2; i++) {  
            width[i] = 80;  
        }  
        ArrayList<InputStream> patt = new ArrayList<InputStream>();  
        patt.add(getResources().openRawResource(R.raw.patthiro));  
        patt.add(getResources().openRawResource(R.raw.pattkanji));  
        artToolkitDrawer = new ARToolkitDrawer(camePara, width, patt, mRenderer);  
    }  
    // mMediaPlayer = MediaPlayer.create(this, R.raw.miku_voice);  
    // mMediaPlayer.setLooping(true);  
    // mMediaPlayer.setOnPreparedListener(new MediaPlayer.OnPreparedListener() {  
    //     public void onPrepared(MediaPlayer mediaPlayer) {  
    //         artToolkitDrawer.setMediaPlayer(mediaPlayer);  
    //     }  
    // });  
    // }  
    FrameLayout frame = (FrameLayout) findViewById(R.id.frame);  
    mGLSurfaceView = new GLSurfaceView(this);  
    mGLSurfaceView.setEGLConfigChooser(8, 8, 8, 8, 16, 0);  
    mGLSurfaceView.getHolder().setFormat(PixelFormat.TRANSLUCENT);  
    mGLSurfaceView.setZOrderOnTop(true);  
    mGLSurfaceView.setRenderer(mRenderer);  
    frame.addView(mGLSurfaceView);  
}
```



Espero que con esta pequeña explicación entiendan un poco como modificar el ejemplo, yo hice el cubo de la exportación .md2 para processing que tiene Emilio en su canal y funciona correctamente.